STATE OF CALIFORNIA • DEPARTMENT OF TRANSPORTATION

TECHNICAL REPORT DOCUMENTATION PAGE

TR0003 (REV 10/98)

ADA Notice

For individuals with sensory disabilities, this document is available in alternate formats. For information call (916) 654-6410 or TDD (916) 654-3880 or write Records and Forms Management, 1120 N Street, MS-89, Sacramento, CA 95814.

1. REPORT NUMBER
CA14-2338

2. GOVERNMENT ASSOCIATION NUMBER

3. RECIPIENT'S CATALOG NUMBER

4. TITLE AND SUBTITLE
Continued Evaluation of Pothole Patching Equipment, Materials, and Processes

5. REPORT DATE
06/14/2014

6. PERFORMING ORGANIZATION CODE

7. AUTHOR
Duane Bennett, Steven A. Velinsky

8. PERFORMING ORGANIZATION REPORT NO.
UCD-ARR-14-06-30-03

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Advanced Highway Maintenance and Construction Technology (AHMCT) Research Center
Department of Mechanical & Aerospace Engineering
University of California, Davis
Davis, California 95616-5294

10. WORK UNIT NUMBER

11. CONTRACT OR GRANT NUMBER
IA 65A0416, Task ID: 2338

12. SPONSORING AGENCY AND ADDRESS
California Department of Transportation
PO Box 942873, MS #83
Sacramento, CA 94273-0001

13. TYPE OF REPORT AND PERIOD COVERED
Final Report
July 2012 - June 2014

14. SPONSORING AGENCY CODE
Caltrans

15. SUPPLEMENTARY NOTES

16. ABSTRACT

California Department of Transportation (Caltrans) maintenance crews repaired approximately 75,000 potholes from 2012-2013. Widely disbursed over 50,000 lane miles of highway, these potholes in most cases rendered conventional lane closures for safe access impractical. Instead, maintenance crews were typically resigned to capitalize on breaks in traffic to rapidly go out and place cold patch asphalt into a pothole and quickly retreat. After the deaths of two Caltrans workers who were patching potholes in 2006-2007, Caltrans tasked the Advanced Highway Maintenance and Construction Technology (AHMCT) Research Center with developing a safer and more efficient means of patching potholes. The implemented solution involved developing and deploying customized, semi-automated pothole patching equipment, which enabled Caltrans workers to patch potholes from inside the vehicle. After an extensive industry search, the Python Manufacturing Incorporated PHP 5000 hot asphalt patching machine was selected because patch times averaged less than 30 seconds, which denotes a key prerequisite for conducting moving lane closure operations on mainline highways. The Caltrans District 4 (D4) San Jose maintenance station successfully placed an upgraded Python PHP machine into service patching potholes on mainline highways exclusively in moving closures. The Python PHP machine patching operation consistently applied five tons of hot asphalt patches in a couple of hours, filling potholes across several highways throughout the entire D4 region. Utilizing the Python PHP in mainline patching operations helped Caltrans completely eliminate workers' traffic exposure in highway pothole patching operations. Caltrans D4 personnel therefore fully integrated the Python PHP machine into their work schedule. However, the Python PHP machine was removed from service due to safety and handling concerns expressed by Caltrans Headquarters Maintenance. As a result of the Python PHP's removal from service, the AHMCT Research Center received notification to stop further research on a supplemental task to add urethane spall patching to the Python PHP machine. The Python PHP machine remains functional and located at the University of California, Davis. Meanwhile, Caltrans D4 workers returned to patching potholes manually on foot until improved equipment or other alternatives become available.

17. KEY WORDS
Semi-Automated pothole patching

18. DISTRIBUTION STATEMENT
No restrictions. This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161.

19. SECURITY CLASSIFICATION *(of this report)*
Unclassified

20. NUMBER OF PAGES
72

21. COST OF REPORT CHARGED

Reproduction of completed page authorized.

DISCLAIMER

The research reported herein was performed by the Advanced Highway Maintenance and Construction Technology (AHMCT) Research Center, within the Department of Mechanical and Aerospace Engineering at the University of California – Davis, for the Division of Research, Innovation and System Information (DRISI) at the California Department of Transportation. The AHMCT Research Center and DRISI work collaboratively to complete valuable research for the California Department of Transportation.

The contents of this report reflect the views of the author(s) who is (are) responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the STATE OF CALIFORNIA or the FEDERAL HIGHWAY ADMINISTRATION. This report does not constitute a standard, specification, or regulation.

The contents of this report do not necessarily reflect the official views or policies of the University of California. This report does not constitute an endorsement of any products or services described herein.

For individuals with sensory disabilities, this document is available in Braille, large print, audiocassette, or compact disk. To obtain a copy of this document in one of these alternate formats, please contact: the Division of Research, Innovation and System Information, MS-83, California Department of Transportation, P.O. Box 942873, Sacramento, CA 94273-0001.

Advanced Highway Maintenance and Construction Technology Research Center

Department of Mechanical and Aerospace Engineering
University of California at Davis

Continued Evaluation of Pothole Patching Equipment, Materials, and Processes

Duane Bennett: Development Engineer
Steven A. Velinsky: Principal Investigator

Report Number: CA14-2338
AHMCT Research Report: UCD-ARR-14-06-30-03
Final Report of Contract: 65A0416, Task ID: 2338

June 14, 2014

California Department of Transportation

Division of Research, Innovation and System Information

Abstract

Caltrans maintenance crews repaired approximately 75,000 potholes from 2012–2013. Widely disbursed over 50,000 lane miles of highway, these potholes in most cases rendered conventional lane closures for secure access impractical. Instead, maintenance crews were typically resigned to capitalize on breaks in traffic to rapidly go out and place cold patch asphalt into a pothole and quickly retreat. After the deaths of two Caltrans workers who were patching potholes in 2006-2007, Caltrans tasked the Advanced Highway Maintenance and Construction Technology (AHMCT) Research Center with developing a safer and more efficient means of patching potholes. The implemented solution involved developing and deploying customized, semi-automated pothole patching equipment, which enabled Caltrans workers to patch potholes from inside the vehicle. After an extensive industry search, the Python PHP Manufacturing Incorporated PHP 5000 hot asphalt patching machine was selected because patch times averaged less than 30 seconds, which denotes a key prerequisite for conducting moving lane closure operations on mainline highways. The Caltrans District 4 (D4) San Jose maintenance station successfully placed the enhanced Python PHP machine into service patching potholes on mainline highways exclusively in moving closures. The Python PHP machine patching operation consistently applied five tons of hot asphalt patches in a couple of hours, filling potholes across several highways throughout the entire D4 region. Utilizing the Python PHP in mainline patching operations helped Caltrans completely eliminate workers' traffic exposure in highway pothole patching operations. Caltrans D4 personnel therefore fully integrated the Python PHP machine into their work schedule. However, the Python PHP machine was removed from service due to safety and handling concerns expressed by Caltrans Headquarters Maintenance. As a result of the Python PHP's removal from service, the AHMCT Research Center received notification to stop further research on a supplemental task to add urethane spall patching to the Python PHP machine. The Python PHP machine remains functional and located at the University of California, Davis. Meanwhile, Caltrans D4 workers returned to patching potholes manually on foot until improved equipment or other alternatives become available.

Table of Contents

LIST OF FIGURES

List of Acronyms and Abbreviations

Acronym	Definition
AC	Asphalt Concrete
AHMCT	Advanced Highway Maintenance and Construction Technology
ATIRC	Advanced Transportation Infrastructure Research Center
CALTRANS	California Department of Transportation
CAN	Controller Area Network
CARB	California Air Resources Board
CFR	Code of Federal Regulations
D3	District 3
D4	District 4
DOT	Department of Transportation
EPA	Environmental Protection Agency
GVW	Gross Vehicle Weight
I	Interstate
IMMS	Integrated Maintenance Management System
L	Liter
MAZEEP	Maintenance Zone Enhanced Enforcement Program
MD/GC	Meter Dispensed/Gravel Extended Kit
META	Maintenance Equipment Training Academy
Mfg.	Manufacturing
MTAG	Maintenance Technical Advisory Guide
PCC	Portland Cement Concrete
PHP	Pothole Patcher
TMA	Truck Mounted Attenuator
UC-Davis	University of California, Davis
Vdc	Direct Current Voltage
VIN	Vehicle Identification Number

Executive Summary

Caltrans maintenance crews repaired approximately 75,000 potholes from 2012–2013. Widely disbursed over 50,000 lane miles of highway, these potholes in most cases rendered conventional lane closures for safe access impractical. Instead, maintenance crews were typically resigned to capitalize on breaks in traffic, either natural or created by California Highway Patrol assistance, to rapidly go out and place cold patch asphalt. This type of patching practice consists of casting a volume of cold patch asphalt into a pothole, compacting it a couple of times with a shovel or boot, and quickly retreating from the roadway. After two consecutive Caltrans worker deaths in 2006-2007 while engaging in this activity, Caltrans tasked the Advanced Highway Maintenance Construction and Technology (AHMCT) Research Center with developing a safer and more efficient means of patching potholes. The implemented solution involved developing and deploying semi-automated pothole patching equipment, which enabled Caltrans workers to remain inside the vehicle while patching potholes. The Python PHP Manufacturing Incorporated PHP 5000 hot asphalt patching machine was selected because it features the potential for the quickest patch application time, which is a prerequisite for safely conducting moving lane closure operations on mainline highways. The AHMCT Research Center, in partnership with Caltrans maintenance crews, developed modifications to the Python PHP machine and introduced innovative procedures, which together reduced the average mainline patch application times to less than 30 seconds. The Caltrans District 4 (D4) San Jose maintenance station successfully placed the adapted Python PHP machine into service patching potholes on mainline highways exclusively in moving closures. The Python PHP machine patching operation consistently applied five tons of hot asphalt patches in a couple of hours, filling all potholes across several highways spread over a large geographical area. The Python PHP machine crew operational area covered the entire D4 region with local crews providing traffic control and pre-marking potholes to be patched. The Python PHP patching operation provides the means to completely eliminate the traffic exposure of Caltrans workers when patching potholes by conducting preventative maintenance operations to eliminate or minimize the development of large potholes that require immediate attention.

Caltrans subsequently funded the AHMCT Research Center's work to extend the capabilities of the Python PHP machine, including dispensing more appropriate materials for Portland Cement Concrete (PCC) spall repair and production data collection. The Caltrans

Maintenance Technical Advisory Guide (MTAG) recommended against bituminous asphalt as a PCC spall repair material due to its dark appearance, which may easily be mistaken as a driver avoidable hazard. Enhancing the Python PHP machine to include the application of a Caltrans approved urethane patch or similar material would enable crews to place a permanent spall patch rather than just a temporary asphalt patch. A fast setting urethane would be the most appropriate epoxy material to support the Python PHP moving lane closure operation, and the urethane dispensing equipment could be readily adapted for use on the Python PHP machine. The AHMCT Research Center researchers completed a Python PHP urethane dispensing adaption design; however, the AHMCT Research Center received notification to stop further research prior to fabrication. Additionally, the AHMCT Research Center installed a PreCise data-logging unit on the Python PHP machine to enable real-time, web-based monitoring of the Python PHP patching operations. The PreCise unit tracks the Python PHP location and critical inputs, which indicate the machine’s patching time. The PreCise website collects machine data and provides tools to produce detailed operational reports.

Background

Caltrans maintenance crews repaired approximately 75,000 potholes disbursed over 50,000 lane miles of highways from 2012–2013 [1]. Potholes emerge sporadically on the highway and rarely occur in concentrated areas. Their characteristic disbursement makes scheduling mainline traffic lane closures for safe access impractical. Therefore, maintenance crews generally rely on traffic breaks with or without California Highway Patrol assistance, depending on traffic flow, to apply quick patches. Maintenance workers resort to quickly moving out to a pothole on the mainline during a brief traffic break, casting a volume of cold patch asphalt into the pothole, compacting it a couple of times with a shovel or boot, and quickly retreating from the roadway (see Figure 1). Limiting pothole repairs to those in need of immediate attention also reduces the risk of exposing workers on foot to traffic. However, fewer pothole repairs adversely affect the public's perception of Caltrans' level of service. After the deaths of two Caltrans workers filling potholes on foot in 2006-2007, Caltrans tasked the Advanced Highway Maintenance and Construction Technology (AHMCT) Research Center to find a safer and more efficient means of patching potholes. The logical solution entailed developing an alternative method for Caltrans workers to respond to urgent highway pothole repairs without exposing themselves to traffic. The most favorable means of achieving this goal was the deployment of self-contained equipment, which semi-automates the pothole patching process and enables the workers to apply patches remotely from inside the vehicle's cab. Ideally, operation of a semi-automated pothole patching machine would not compel any workers to be on foot on the highway to support the patching process. The total elimination of worker traffic exposure while patching on the highway delivers Caltrans the maximum possible worker safety gain, which provides strong justification for the added costs of owning several relatively expensive, semi-automated pothole patching machines.

Over the years, several companies have developed different versions of semi-automated pothole patching machines; some have gone on to become commercially available, yet only a few of those have earned any level of enduring support from a state Department of Transportation (DOT). These types of machines typically dispense either hot asphalt or an emulsion-based spray patch. While both processes provide benefits, the hot asphalt process remains the most traditional and, therefore, the least controversial. Additionally, the quicker application of the hot asphalt patch is a distinct advantage over the far slower spray patching method when operating in moving traffic closures on mainline highways, where application speed is critical. However, the spray patch method’s main advantage is its ability to store patching materials onboard the spray patch vehicle for quick and effective deployment on both large- and small-scale jobs. In contrast, the hot asphalt patch process requires purchasing and loading hot asphalt from a batch plant for one use during a workday; any remaining unused asphalt must be purged from the machine at the end of the workday. The machine depicted in Figure 2 uses a spray patch based approach, whereas Figure 3 shows a hot asphalt machine in operation on a California highway. Since, anecdotally, the durability of both types of patches is rated equally well, the selection that best suited Caltrans’ patching needs was based on the differences in the specific mechanics of each patching process. Implementing semi-automated equipment based on either of these processes would require a major evolution in the conventional Caltrans maintenance patching procedures. The research involved considering and comparing an ideal patching operational scheme for each process that was suitable for Caltrans operations to determine which machine achieved the project goal of completely eliminating worker traffic exposure.

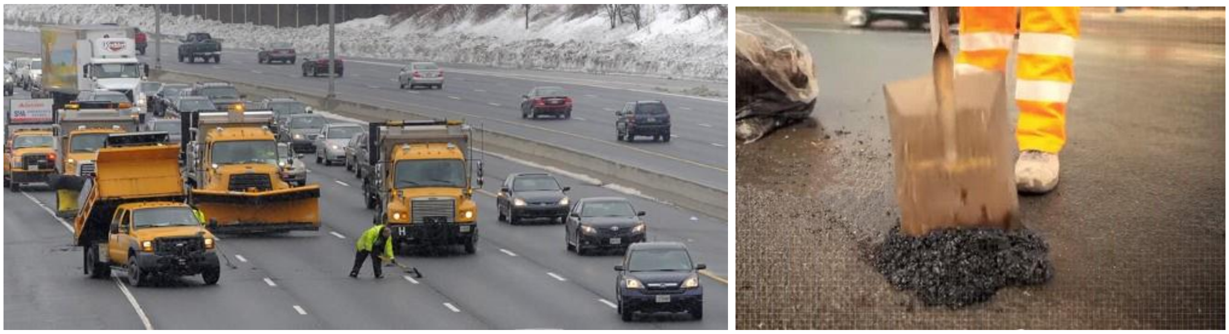


Figure 1. Manual Pothole Patching on Mainline Highway

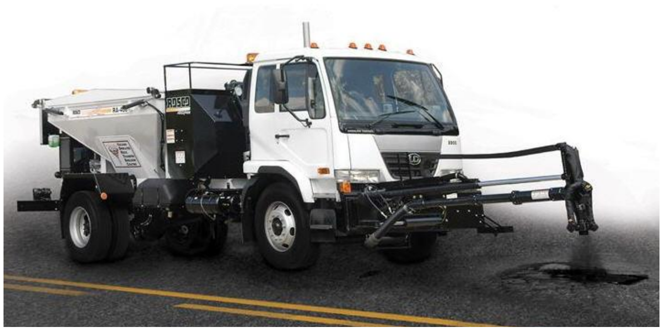


Figure 2. Rosco Spray Patch Machine

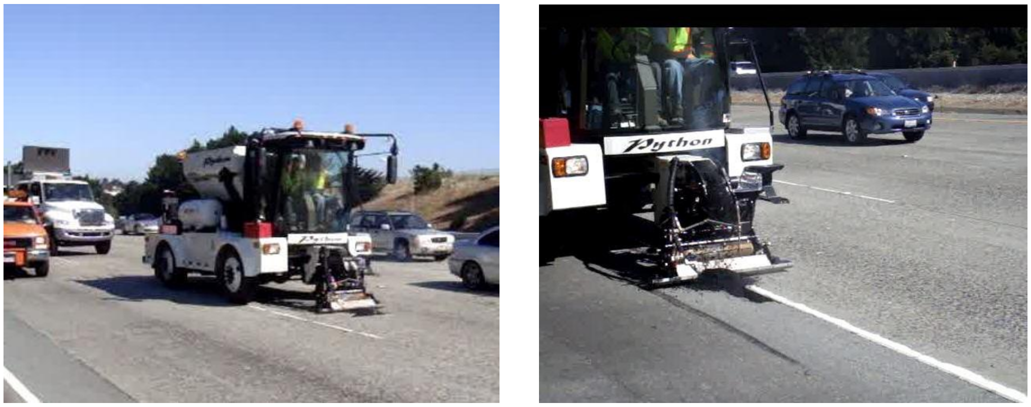


Figure 3. Python PHP Asphalt Patching Operation – Moving Closure on Interstate (I) 280

Semi-Automated Pothole Patching Operational Issues

Caltrans mainline pothole patching operations are frequently small-scale, ad-hoc, quick response actions to either motorist complaints or patrolling Caltrans worker notifications. Once identified, Caltrans is accountable for mitigating any imminent hazards without delay, which often translates into patching by whatever means are at hand and Caltrans workers patching potholes without lane closures. However, the application of expensive, vehicle-based, semi-automated pothole patching systems to this type of quick response pothole repair is problematic for Caltrans.

Equipment Cost

Semi-automated patching equipment is too expensive to be widely distributed in regional maintenance yards. An optimistic projection might be one machine for each Caltrans District, but more likely, rural districts would share units. Multiple units in a single district are also unlikely since semi-automated patching equipment has very high patching rates and can easily maintain a large geographic area. The practice of utilizing pooled vehicles, a common technique of Caltrans, maximizes equipment usage efficiencies, but the success of this approach depends on the advanced planning of a task. This is not to say that Caltrans does not also conduct planned pothole patching operations that would benefit from the pooled patching machines scheme, just that this approach would not in any likelihood achieve the goal of the total elimination of workers' traffic exposure. The delays and extra work associated with scheduling pooled patching machines would often result in Caltrans workers taking the easier route of risking their traffic exposure to make quick manual repairs.

Dedicated Operator

Semi-automated patching equipment is fairly complex to operate and can also be demanding to service and maintain in the field. The additional costs associated with operating semi-automated patching machines must be accounted for and managed effectively, or the net overall cost benefits can easily turn negative. As an essential investment step, restricting machine operation to fully trained, dedicated operators maximizes production benefits and minimizes operational costs. Skilled operators can typically become proficient with basic semi-automated patching machine operation after only a few hours of instruction. However, the skill to patch potholes efficiently with semi-automated equipment cannot be simply taught; it is only developed in time through hands-on experience. Experience notably correlates to the speed of the patch application, a critical factor in successfully conducting mainline highway patching operations. Experienced operators additionally apply their own successful and sustainable approaches to clean-up procedures and service techniques.

Service Delay

A strict cleaning regimen must be followed after each operation of semi-automated patching equipment regardless of the machine type utilized. Whether emulsion-based or hot asphalt based, passageways and surfaces must be cleaned thoroughly to avoid clogs and the buildup of notoriously sticky patching materials. Cleaning time combined with the time to refill patching materials and the time to complete standard, daily service requirements can all add up to a relatively large overall time investment when patching only a few potholes. This comparatively protracted machine service delay results in a negative incentive for workers to utilize the semi-automated machine and instead just continue on with traditional, manual methods of asphalt patching for spot repairs.

Caltrans Semi-Automated Pothole Patching Operational Plan

Semi-automated pothole patching machines, by design, typically operate on secondary roads or two-lane highways. Unfortunately, the potholes on the highway mainline pose the greatest threat to Caltrans workers. Semi-automated patching machine operations have been successfully demonstrated on the highway mainline, but due to inherently lengthy patch times, these were conducted in traditional, stationary, temporary lane closures. When considering multiple lane highways, these stationary closures pose problems including closing multiple lanes and ramps, requiring permission from traffic operations, and posting signage days in advance. As already discussed, potholes occur sporadically, so a single stationary closure would probably include just one major pothole and possibly a couple of minor potholes. Consequently, one to two stationary closure highway patching operations a day would be expected to fill approximately two to six mainline potholes regardless of the style of semi-automated patching utilized. Mainline stationary lane closure patching operations are overly constricting and inefficient, which is the reason traffic break manual patching operations still persist despite the perils of direct traffic exposure. An optimal solution incorporates the utilization of in-cab operated patching equipment in moving closure patching operations on the highway. Vehicle-based, moving closure patching operations can allow maintenance personnel to patch a relatively very high number of mainline potholes, eliminate all worker exposure to direct traffic, and patch difficult to access middle lanes and ramps without rerouting traffic. Highway moving closure, semi-automated patching operations include two constraints: first, the patching process must be quick, in half a minute or less; secondly, trailing high-speed traffic must be able to drive over the patch immediately without the material rutting or emitting debris that could potentially damage vehicles.

The implementation of a safe and efficient traffic management scheme tailored for moving lane closure operations is essential when deploying semi-automated patching equipment onto the highway mainline. Simply purchasing and placing the equipment in the hands of an experienced equipment operator fails to ensure workers will realize the machines' innovative production potential or embrace the extra effort needed to utilize high-tech, semi-automated patching equipment safely and effectively. As previously discussed, basic operation issues such as pooled equipment and dedicated operators require high-level planning and district-wide coordination, which extends beyond the scope of most equipment operators' responsibilities. Since pothole patching is a low-level, primarily reactionary concern, local managers will need guidance on how their operations can be reorganized to benefit from the use of the semi-automated patching machine. The successful demonstration of significantly increased patching production rates combined with the elimination of all worker exposure (as documented in this report) strongly supports investing the additional expense, effort, and dedicated operator resources.

A series of highway trials in partnership with Caltrans District 4 maintenance crews led to the development of a successful Caltrans deployment/utilization strategy. District 4 crewmembers used the Python PHP Manufacturing Incorporated PHP 5000 machine for these trials, but the utilization strategy should be considered generic to Caltrans for all self-contained patching systems, independent of process type. The patching machine and the dedicated operator are based out of a single yard with access to patching materials. The machine is dispatched to areas in the region where potholes need to be filled. Typically, maintenance crews geographically log and mark the potholes with paint in a separate vehicle days ahead of the patching operation. Then, the supervisor of the area maintenance yard provides traffic control and coordinates with the Maintenance Zone Enhanced Enforcement Program (MAZEEP). The operation forms up on the highway, takes lanes in a moving closure, and drives along to the first marked pothole. As shown in Figure 4, the Python PHP machine stops and fills the pothole, potentially in less than 30 seconds, as the traffic control vehicles continue moving, closing the gap. Before the gap closes, the Python PHP machine finishes a pothole repair and drives along to the next marked pothole, thus reestablishing the traffic control gap. Typically, the moving closure covers two lanes and the shoulder to enable the patching machine to potentially access more potholes in a single pass. When patching across highway ramps, MAZEEP deploys ahead of the patching operation and blocks entry ramps while the patching operation passes. Along with MAZEEP, the patching operation traffic control blocks the exit ramps, thus temporarily closing ramps for a minute or two. However, the off-ramp closure can often be problematic because a few motorists refuse to redirect to the next ramp and attempt to drive through. Caltrans workers are not exposed, but stopping to address the breach in the closure can lead to a traffic collision. Displaying “ramp closed” signage on the Truck Mounted Attenuator (TMA) trucks may help prevent this situation.

The Caltrans District 4 trials involving the Python PHP machine patching in moving lane closures conclusively demonstrated that this type of deployment scheme enables a single crew to fill all the mainline potholes on multiple highways over a large geographic area in a single day. Figure 5 shows a map of a single day’s production, including filling 75 potholes and seven longitudinal cracks in two hours of patching. Most importantly, the high-mobility preventative maintenance aspect of this patching scheme may completely eliminate the necessity for improvised manual patching operations.

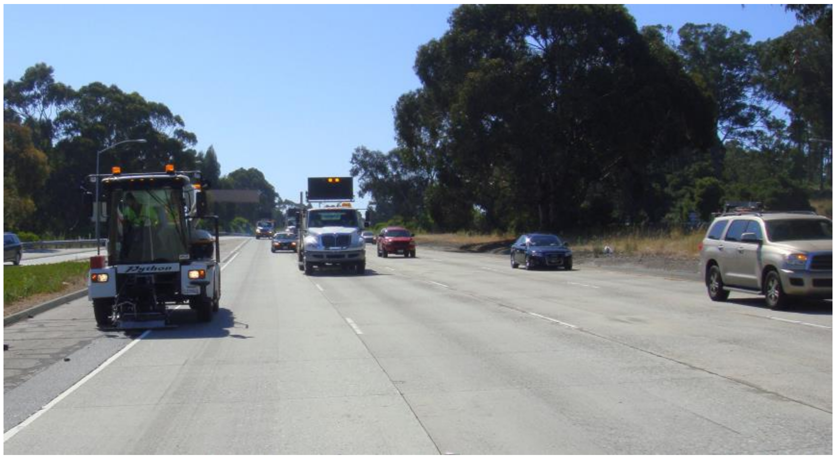


Figure 4. Caltrans Moving Closure Patching Operation

In these highway trials, the District 4 deployment strategy consisted of the semi-automated patching machine traversing the district once or twice a week depending on the demand. Area supervisors would contact the machine’s base of operation, the Caltrans San Jose maintenance station, with patching requests and their priority levels. For urgent requests, the patching machine was routed immediately; less critical patching requests were combined with others in the vicinity along a planned, larger-area, high-production route. With the Caltrans crews in the destination area providing all necessary traffic control resources, the dedicated operator could drive the semi-automated patching machine anywhere in the district in a single day.

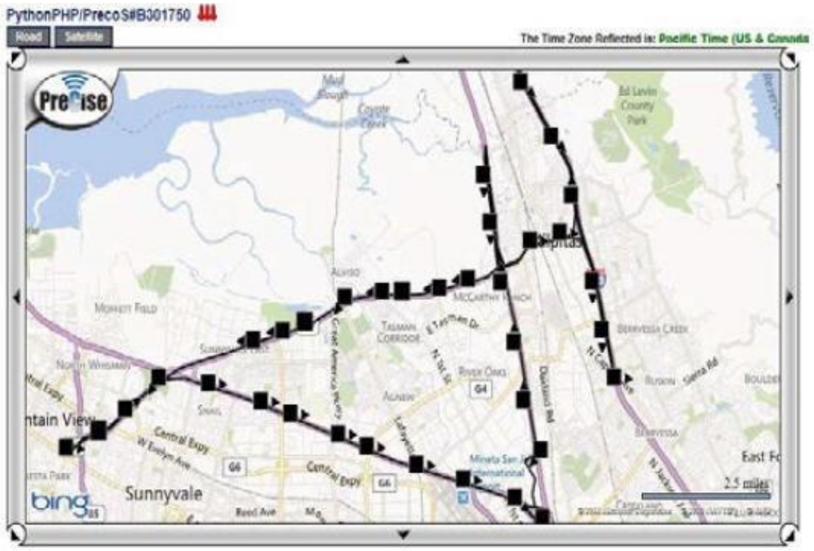


Figure 5. Caltrans District 4 Single Day's Python PHP Patching Data Log

Since patching operations generally occur outside of commute times, the patching machine can easily be prepared for operation within normal working hours and be ready for operation by the end of the morning commute traffic. Areas of the highway that contain either the highest concentrations of potholes or the highest priority repairs are typically focused on first, but the highly mobile nature of the Python PHP operation enables all pavement holes in the entire service area to be patched quickly and efficiently. An expected benefit of the effective deployment of semi-automated patching equipment is the efficient patching of small potholes in a timely manner before these holes deteriorate further into larger potholes. This unique patching scheme enables Caltrans to complete preventative maintenance to control the development of large potholes. As a result, the frequency of pothole complaints by the motoring public can be expected to decrease, thereby improving the perception of Caltrans' level of service.

Examining Semi-Automated Pothole Patching Methods

Conventional pothole patching basically consists of a four-step process:

1) Clean loose debris and dust from the pothole

2) Apply tack spray to improve patch adhesion

3) Fill the pothole with patch material

4) Compact/level patch material in the pothole

Depending on the process, these basic steps are modified in different ways to assist with automation and moving closure patching operations. Fundamentally, two common, vehicle-based processes to patch roadway potholes exist: the spray patch and the hot asphalt patch methods. Both of these methods produce high-quality, long-lasting patches but utilize distinctively different materials and processes. It is not the goal of this research project to study or rate the quality of patching methods. Several readily available, published scientific studies verified the effectiveness of each of these patching processes [2]. This research project focuses on developing the means to enable Caltrans to incorporate semi-automated patching equipment into their maintenance operations, thereby improving worker safety on the highway and increasing pothole patching production regardless of the patching method.

Spray Patch Method

The semi-automated spray patch method involves filling the pothole with an incompressible mixture of small gravel and liquid emulsion binder. Figure 6 displays a photograph of a spray patch pothole repair. The spray patch process utilizes the following modified four-step method to patch potholes:

4-Step Spray Patch Process:

1) Clean pothole with a compressed air blast
2) Apply tack spray
3) Spray patch mixture into the pothole
4) Apply a no-tack top coat

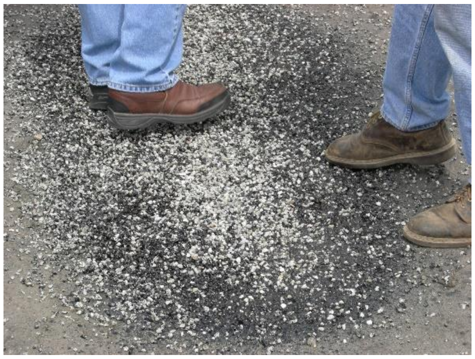


Figure 6. Spray Patch

Replacing the typical mechanical compaction step with high-velocity applied slurry spray to attain some degree of compaction and including the ancillary step of adding a top coating of gravel represent the main changes to this four-step process. The top coating of dry gravel helps prevent the fresh patch from sticking to the tires of the trailing traffic (Figure 7). While this loose gravel topping appears fairly harmless for medium-speed highways and surface roads, the topping runs contrary to Caltrans best practices for high-speed mainline highway maintenance operations. The loose gravel potentially risks being tracked up from the trailing high-speed traffic and chipping windshields and body paint. These risks may be mitigated by possibly modifying the spray patch equipment to apply a crumb rubber material in lieu of gravel.

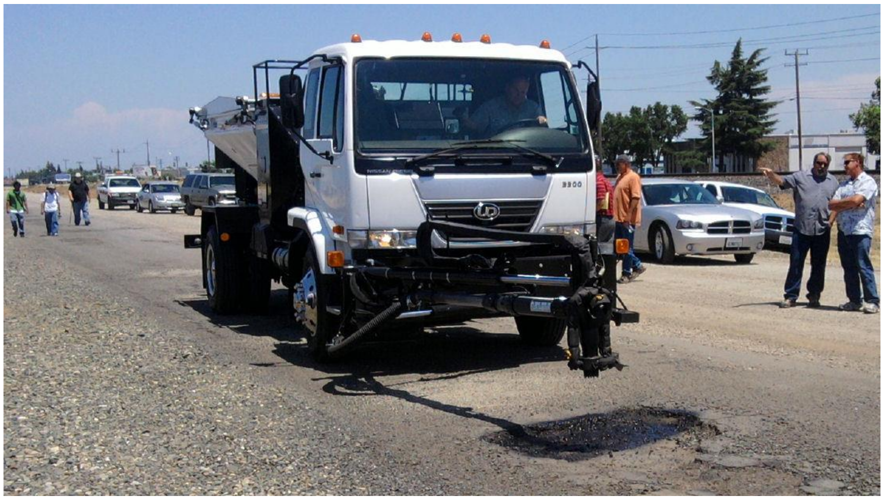


Figure 7. Rosco Spray Patch Machine Demonstration at the Maintenance Equipment Training Academy (META)

Spray patching is fundamentally a slower process than hot asphalt patching since the spray patch machine fills the pothole with patching material in layers built up over many passes. From inside the spray patching vehicle, an operator manipulates the spray nozzle, which is typically mounted at the end of a long hydraulically powered boom. Since semi-automated spray patch equipment lacks secondary tooling or any other type of leveling capability, the patch spray must be applied evenly in the pothole in order to create a relatively flat patch. The process of circling the pothole with the spray nozzle while injecting patching material so it is uniform and level with the roadway consumes most of the time in the spray patch process. In addition, the process of implementing spray patch materials in moving closure patching operations needs to be evaluated. The binder material, an emulsion, needs to dry to become solid. However, this patching material appears stable enough to provide a certain level of resistance to tire rutting almost immediately after application. To better understand if spray patch machines can be modified to reduce patch application time and to overall enhance the support of moving closure operations requires additional research.

The spray patch method’s greatest advantage over the hot asphalt method pertains to material handling. The emulsion and gravel patching materials can be stored in hoppers and tanks on the machine ready for use. This process enables the efficient deployment of the spray patch machine even for small jobs and with short notice, whereas the hot asphalt patching machine needs to be loaded at an asphalt batch plant that may not be local and will have a standard one-ton asphalt minimum sale policy. Once the hot asphalt is loaded into the machine, the material must also be kept heated, and any remainder must be dumped out of the machine at the end of the day's operation. The spray patch materials remain at ambient temperatures with the patching gravel stored on-board the machine. The emulsion, however, must be pumped out of the delivery hose and nozzle into a bucket and replaced with a cleaning solvent to prevent clogging when the spray patching machine is not in use.

Hot Asphalt Patch Method

Currently, only one commercially available, vehicle-based machine, the Python PHP machine, automates the following traditional hot asphalt patching process to repair potholes:

4-Step Hot Asphalt Process:

1) Clean pothole with air jet blast
2) Apply tack oil spray
3) Fill the pothole with hot asphalt
4) Compact the patch with the compaction roller

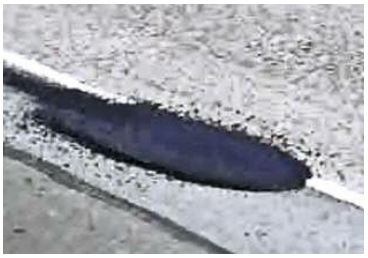


Figure 8. Hot Asphalt Patch

The patch can be applied very quickly, typically in less than a minute, within a moving closure. For in-lane moving closure patches, where patching time is most critical, pothole repair crews often skip the cleaning and tack oil steps unless the pothole contains water or debris. The resulting two-step process further reduces patching time to less than half a minute seemingly without diminishing the longevity of the resulting patch based on Caltrans Maintenance initial Python PHP patching operation observations on Interstate 280. Patch time can be even further reduced with a type of rolling patch application method, whereby the machine stops long enough just to fill the pothole with hot asphalt and lower the compaction roller to the pavement with the rake at compaction level and then compacts the patch as the machine rolls away. Rolling patches are not as sightly but can be typically applied in as little as 15 seconds (Figure 8). After the compaction of the hot asphalt in the pothole, the patch is at full strength and can be driven on immediately by the trailing traffic, providing additional compaction and strength.

Semi-Automated Patching Process Selection

The AHMCT Research Center conducted a detailed survey of commercially available pothole patching machines, which automate either type of patching process. Since the main research objective focused on developing moving closure patching capability, equipment needed to possess the features contained in the following list to be considered for the Caltrans trial program.

- Patching times quick enough for it to safely operate within a moving lane closure
- Vehicle-based, self-contained patching system
- Operator controls entire patching operation from inside the cab
- All support functions monitored and controlled from inside the cab
- Multi-axis application arm providing full range of motion in front of the vehicle
- Clear visibility for driver/operator
- Machine travels at highway speeds to and from job site
- Minimum 3-ton (6,000 lb.) patching material capacity
- Meets California vehicle emissions regulations

The selection of which patching method best suited the Caltrans moving closure patching scheme employed in this research was based primarily on two factors: the quickest patch application time and the patching method's general degree of acceptability. The Python PHP machine was selected as the commercially available machine that best matched Caltrans requirements for this research trial [3].

Python PHP Machine Configured for Caltrans Trial

The Python PHP 5000 pothole patching machine is a self-contained, hydrostatic drive vehicle that can travel at highway speeds. The vehicle's unconventional frame is built around a front-mounted, telescoping, and rotating application arm that resides under a glass-bubble-style harvester cab. The driver/operator is situated in an ideal position for maximum pothole patching visibility directly above and in front of the patching arm. Also, the driver's seat, centered in the vehicle cab, delivers an equally close-up view anywhere in the traffic lane. The Python PHP's unique design optimizes the driver/operator viewpoint with a direct view as close as six feet from the pothole. As a comparison, spray patch machines typically use a truck-based chassis, have a front bumper mounted arm that places the pothole at least 15 feet away from the operator, and positions the operator on the left side of the vehicle cab. In addition, the Python PHP application arm retracts into the machine, and the patching vehicle fits into a standard-sized parking space. On a typical spray patch machine, the boom awkwardly sticks out ten feet from the front of the vehicle constantly, even when driving in normal traffic conditions.

In 2006, Python Manufacturing (Mfg.) obtained the rights to the pothole patching machine design. In 2008, the AHMCT Research Center purchased one of the first five Python 5000 pothole patcher machines designed and built by Python Mfg. In a short time, Python Mfg. significantly improved and modernized the design of the machine, but the semi-automated process continued to follow the long established conventional pothole patching method. The Caltrans version Python PHP should nevertheless be considered a prototype in a number of respects due to its many innovative machine features. A custom designed auxiliary hopper heating system enabled Caltrans crews to extend the time asphalt can remain in the hopper without dumping. Also developed was an environmentally friendly, citrus solvent cleaning system that eliminated the use of diesel solvent for cleaning. The AHMCT Research Center integrated a PreCise data logging system onto the machine to provide reliable wireless vehicle tracking and equipment monitoring. However, the most significant modification involved incorporating the experimental Cummins California Air Resources Board (CARB) 2007 engine, as required to meet California vehicle emission regulations.

The following list contains the Python PHP features customized for Caltrans:

- On-board, gasoline-powered, 13,000 Watt generator
- Auxiliary electrical hopper heating system (overnight)
- Modification to pumps, seals, and hosing to accommodate citrus solvent cleaner
- Spiral brush material delivery belt cleaning attachment
- Environmental Protection Agency (EPA) 2007 and CARB 2007 certified Cummins 6.7 L diesel engine
- Integrated PreCise tracking and data collection system
- Hydraulic pressure-controlled roller to provide consistent patch compaction

The special features included in the Python PHP machine merely benefited Caltrans' unique operational requirements. The customized features maintained the original Python PHP machine patch application process. Special training and innovative, unconventional patch application techniques significantly reduced the patching time necessary to conduct Caltrans moving patching operations.

Python PHP Machine Operator Training

Initially, AHMCT Research Center staff provided Python PHP machine training to Caltrans maintenance equipment operators in partnership with the Caltrans Maintenance Equipment Training Academy (META). The initial Python PHP operator training program consisted of a combination of classroom instruction and equipment operation training. Python PHP machine hands-on patching practice sessions were often conducted at the UC-Davis Advanced Transportation Infrastructure Research Center (ATIRC) facility using portable pothole templates (Figure 9). Caltrans operators also received training internally. After one of the operators sustained an injury during pothole patching operations on the highway, a Caltrans injury investigation was completed. The investigation concluded that the incident was related to operator training. This information supported a decision to consolidate efforts and place training responsibility with the AHMCT Research Center during the research project. Operator training, including safe operation and machine cleaning, is a critical part of a successful machine deployment.

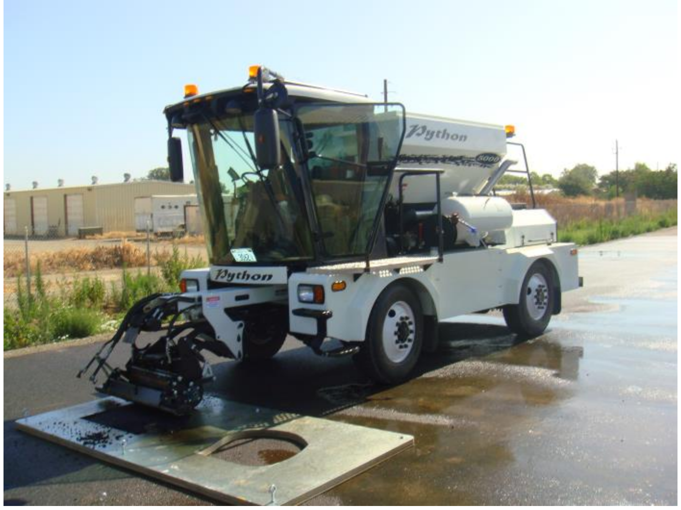


Figure 9. Python PHP Template Training

AHMCT Research Center staff conducted Python PHP operator training on a one-on-one basis at the machine. Individual instruction at the machine proved more effective than the group classroom instruction element. The operator received an instruction manual illustrating the full range of Python PHP control screens as a reference aid. The Python PHP is simple to operate, but only experienced Caltrans equipment operators should be assigned as dedicated Python PHP operators. The experience is required mainly for their decision-making competence when dealing with maintenance operations logistics on the highway in live traffic. Python PHP training with AHMCT Research Center instructors began with a walk around and a pre-operational check pointing out to the operator the important operational features and the overall functionality concept of the Python PHP machine. Next, the training moved into the patching vehicle's cab where the trainer introduced all of the machine controls and screen settings on the Python PHP in-cab control panel to the operator. Then, the operator started the machine and ran all of the machine's components including the patching application head while in the maintenance yard. Once the operator became comfortable with the machine controls, the maintenance supervisor scheduled a patching operation on the highway, including traffic control for a moving closure. The AHMCT Research Center instructor rode along with the operator on their first trip to an asphalt batch plant and then to the traffic control operation on the highway to begin patching operations. The instructor then oversaw the operator during hands-on patching operations on the highway filling potholes in moving closures.

A new operator at the controls of the Python PHP machine naturally focuses on creating the perfect patch, unnecessarily and significantly lengthening the patch application time. Therefore, the AHMCT Research Center's version of Python PHP moving closure training instead focused on fast and efficient pothole patching skills, which yielded very fast patch times. In highway trials working with Caltrans dedicated operators, the AHMCT Research Center developed an innovative method of advancing the Python PHP machine during the compaction and rake stage to shorten patch times. In-vehicle training with the operator at the controls of the Python PHP machine proved most effective as a technique to instruct operators on mainline moving closure pothole patching. The AHMCT Research Center instructor rode along as needed to offer suggestions and answer questions during the operation, but each operator quickly developed their own unique patching style. Typically, after just a couple of patching training operations, the operators established a personal style and comfortably patched potholes independently.

Cleaning and maintenance remain important aspects of the operator training necessary to keep the Python PHP machine in service and operating efficiently. Python PHP machine cleaning actually began prior to running the hot asphalt through the system by spraying a thin coating of citrus cleaner on all patching head surfaces and the material belt. This practice helped prevent hot asphalt from building up during operation and assisted with the post-operational cleaning. During the Python PHP machine post-operational procedures, the operator generally drove the Python PHP to a Caltrans off-site materials storage area, often adjacent to the highway, to remove any excess asphalt tailings before returning to the maintenance yard. The asphalt capacity, usually the patching operation's limiting factor, normally led to an empty hopper at the end of the patching operations. The operator removed any remaining asphalt out of the machine's hopper and any asphalt build-up on the patching surfaces and material belt with a jet spray of citrus cleaner and a long handle scraper. The AHMCT Research Center instructor oversaw and instructed the new operator on how to configure the machine for post-operational cleaning with the arm out, hopper open, and the material belt running. With most of the asphalt removed, the equipment operator returned the Python PHP machine to the Caltrans field station to complete the post-operational cleaning with water and citrus cleaner on the maintenance yard's wash rack. The operator used a jet of water from a hose, citrus cleaner solvent, and possibly a brush to remove asphalt residue from the exterior of the Python PHP machine and all patching surfaces. In addition, the clean-up procedure involved squirting water into the hopper to flush out any remaining gravel and fine residue as the material belt ran. After the dedicated operator cleaned the patching machine, he or she typically refueled the vehicle, filled the grease points, and inspected the Python PHP machine in preparation for the next patching operation.

Python PHP Machine Trial Deployment

The Python PHP machine was initially deployed in the District 3 (D3) Sunrise maintenance station, but the highways in their area lacked the type of traditional potholes targeted for this deployment trial. A number of the D3 potholes actually resulted from failed areas of thin asphalt concrete overlay on PCC highways, which delaminated into large patches. These types of failures are not suitable for patching by the Python PHP operation. The Python PHP needed to be relocated to a region with a large number of conventional potholes in need of repair. In addition, the Python PHP machine suffered from some technical problems during initial testing and deployment. A standalone section later in this document titled "Python PHP Machine Deployment Issues and Upgrades" describes the technical issues in detail.

The Python PHP machine was then moved to the District 4 Foster City maintenance station and utilized to patch primarily PCC spalls on the mainline exclusively in moving closures on Interstate 280 and other highways in their maintenance area. The Python PHP machine remained in service in the Foster City maintenance station area for two months, and this experience enabled Caltrans and the AHMCT Research Center to work in partnership on the highway to develop a fast patching methodology, which reduced patching times to the fifteen-second range. During this process, the Foster City dedicated operator unexpectedly received a promotion to a new position, resulting in the need for another trainable and experienced equipment operator.

The Python PHP machine was then transferred to the District 4 San Jose maintenance station, where an experienced equipment operator was available to be trained as the Python PHP dedicated operator. AHMCT Research Center personnel trained the new operator on the innovative patching techniques developed with the Caltrans Foster City crew, which maximized the operational efficiency. Using these new patching practices in moving closure operations, the Caltrans San Jose crew quickly achieved very high production rates when compared to traditional pothole patching operations. In addition to patching the difficult to reach potholes that Caltrans normally patches, the Python PHP machine enabled crews to patch all pavement holes encountered as a preventive measure. Utilizing the Python PHP machine, the Caltrans San Jose crew patched all roadway pavement holes of any size in the South Bay Area in a few weeks, conducting patching operations once or twice a week. The San Jose crew was in the process of extending their patching operations district wide when Caltrans Headquarters Maintenance expressed safety concerns about the Python PHP machine and removed the unit from service.

The Python PHP machine achieved full deployment status and was independently operated by Caltrans District 4 with the AHMCT Research Center providing only equipment support and maintenance as needed. Due to this successful deployment of the Python PHP machine to Caltrans District 4, the AHMCT Research Center could not obtain firsthand knowledge of the detailed logistics of the San Jose Python PHP patching operations. Since the Python PHP machine deployment prematurely ended during the deployment trial, the goal of generating a detailed analysis of production data could not be achieved. However, between Caltrans Integrated Maintenance Management System (IMMS) data and the onboard PreCise data logger record, the very high patching production rate can be recognized. The appendix of this report contains the AHMCT Research Center’s Python PHP Deployment Summary of some of these operations. Some of the data on these summary reports comes directly from the PreCise data logger. The data generated from the PreCise data logger includes maps displaying the Python PHP machine's patching routes as well as illustrations of the overall lane miles patched at two lanes per pass while patching all pavement potholes encountered. Besides the information from the PreCise data logger, the summaries also contain additional information, including observatory and operational notes, the number of potholes/spalls filled on some of the latter summary reports, and IMMS work order references.

Caltrans Python PHP Patching Operation Case Study

A typical Python PHP patching operation begins on the day prior to highway operations by contacting the nearest asphalt batch plant in the work area to determine the availability of hot asphalt. If asphalt is available, then the maintenance crew requests a work zone placement with traffic operations and MAZEEP. The day of patching begins with the operator driving the Python PHP machine to the batch plant and loading the machine's hopper with hot asphalt. The loading process is simply a matter of driving the Python PHP machine under the asphalt plant chute and opening the hopper doors to allow the desired amount of asphalt to drop directly into the machine hopper. The operator then drives the Python PHP machine to the work zone, where traffic control awaits nearby. The traffic control operation takes the lanes, and the operator pulls the Python PHP machine out on the lanes and drives along, stopping briefly to patch potholes and then accelerating quickly to the next pothole. On multilane highways, the patching operation typically extends across two lanes for each pass. MAZEEP briefly holds traffic on ramps and releases the traffic hold as the Python PHP machine passes. Due to the high rate of speed that the Python can travel between potholes, it is difficult for the operator to see far enough ahead to line-up the machine for the next pothole patch. It is therefore preferred that the potholes be identified with a visible paint mark on the shoulder prior to initiating the patching operations. Generally, the PHP dispenses the hot asphalt in a couple hours during a patching operation, and the traffic control resources are dispatched to other tasks for the remainder of the day. The operator drives the Python PHP machine to a maintenance yard for post-operation cleaning and light machine maintenance to prepare the Python PHP machine for the next patching operation, which is typically conducted within a few days.

The Python PHP moving closure patching process creates safe access to nearly all potholes on the highway. Potholes appear most often in areas with the highest traffic volumes and are often inaccessible to Caltrans workers utilizing manual patching methods due to line of sight limitations, traffic merges, and other reasons. Gaining access to potholes in these areas is more involved and requires a stationary lane closure. Fixed closures often require detours, closure postings, weekend or night work, and, in some cases, complete highway closures. The Python PHP moving patching operation provides Caltrans the ability to patch potholes that would otherwise not be reachable to repair using current methods or that would require emergency, high traffic impact, or complex planning actions to facilitate access.

Python PHP Machine Cost / Benefit Information

Developing a successful utilization strategy that would provide Caltrans with a positive cost benefit presented a primary challenge for the AHMCT Research Center. Such a strategy would need to overcome the major obstacles of both the high cost of semi-automated machinery and the sporadic distribution of potholes on the highway. Establishing the cost benefit comparison analysis between manual patching and semi-automated patching poses a challenge since their resulting patches are fundamentally different. The conventional pothole patching task consists mainly of a manual operation requiring only a shovel and a sack of cold mix and provides only a temporary patch. In contrast, a semi-automated pothole patching machine requires a large capital investment and produces a semi-permanent patch. It is still critical to develop a cost savings model because even significant improvements in worker safety, pothole accessibility, and patch quality provided by automation alone will not drive the demand for adopting this sophisticated equipment in the fleet if the cost of patching remains unchanged or increases. The best means of guaranteeing the widespread acceptance of automation entails that the use of the equipment dependably delivers a large bottom-line cost savings to Caltrans. Since the direct costs of the asphalt patch material itself are similar for both hand-applied and semi-automated methods, only a substantial increase in the patch production rate could deliver the necessary bottom-line cost savings.

Adapting the Python PHP machine for moving closure patching operations was determined to be the best strategy to achieve the necessary production boost. Generally, inhibited access to spalls or potholes currently limits the pothole patching production of Caltrans crews. The need to establish fixed temporary lane closures or traffic breaks to conduct conventional on-foot (and exposed to traffic) manual pothole patching operations on mainline highways is problematic. In addition, all currently available commercial, semi-automated patching machines, including the Python PHP, are designed for pothole patching in a stationary patching operation, on secondary roads, or on highways where temporary fixed work closures are possible. Creating the moving closure capability therefore became essential due to Caltrans Division of Maintenance's most pressing need for mainline pothole patching.

In collaboration with Caltrans District 4, AHMCT Research Center maintenance personnel developed Python PHP machine modifications and procedures that allowed for moving mainline patching operations in live traffic lanes. This innovative operation mitigated the pothole access issue and opened the door for unprecedented patching production opportunities. Operating the Python PHP machine once or twice a week in moving closure mainline patching operations, Caltrans District 4 San Jose crew routinely applied 9,000 lbs. of hot asphalt patches in under two hours on the highway. Normally, this crew only responds to patch potholes when prompted by motorist complaints or when significant potholes are identified by Caltrans and applies, on average, less than 100 lbs. of patching material per day. This significant increase in production, however, does not translate into a cost benefit if the perception of Caltrans level of service remains unchanged. Continuing the practice of patching a few potholes per day but patching them incrementally faster with the Python PHP machine proves ineffective. Instead, the real cost value of the Python PHP can be realized when the level of service extends to filling all holes of appreciable size on the mainline highway. In the Python PHP operation, the crew not only patched the potholes that were of immediate concern, but also repaired all holes encountered. This preventative maintenance approach helps prevent the growth of these small holes into more significant potholes. The much improved highway infrastructure made possible by the Python PHP provided an increase of level of service difficult to quantify but clearly of value.

The San Jose patching crew using the Python PHP machine routinely applied approximately 90 times more patching material on the highway per day than when using their conventional hand patching operation. With the Python PHP machine, the San Jose crew took approximately seven days to patch all the pavement holes on all the mainline highways in their area of operation. Assuming that the crews would use stationary closures to access all these mainline pavement holes, they would have taken approximately 630 work days (just over two years) to patch these holes by hand. Notably, this unrealistic practice explains why Caltrans maintenance focuses on patching only the holes large enough to be considered a traffic hazard. Some might argue that a manual patching crew could easily place far more than 100 lbs. of patching material per day. This line of reasoning may be true, but similarly, the PHP could easily be refilled with asphalt and place 18,000 lbs. per day. The goal here illustrates the relative magnitude of the increase in productivity. These reported figures are empirically based on Caltrans maintenance personnel interviews and direct observations of Caltrans standard procedures.

A quantitative cost-benefit analysis of the Python PHP should realistically focus on the benefit of level of service improvement and not be based purely on reducing the cost of current methods. According to a recent study, driving on unmaintained roads costs California motorists $44 million annually in the form of additional vehicle operating costs, congestion-related delays, and traffic crashes [4]. Potholes, spalls, and other patchable pavement issues represent major contributing factors to this cost when considering pothole-related collisions and increased vehicle costs. With a small fleet of pooled Python PHP machines, Caltrans could realistically institute a statewide preventative maintenance policy of placing patches on all mainline pavement holes, which would greatly reduce costs to the motoring public in addition to improving worker safety.

Python PHP Machine Deployment Issues and Upgrades

In July 2009, the AHMCT Research Center began operations with the Python PHP machine, and initially the machine functioned as expected without any problems. A few months later, with only minimal use, occasional faulting began to occur, with the Python PHP machine briefly losing engine power while driving on the highway at full power. Soon the faulting occurred fairly consistently, so the manufacturer became involved to fix the problem. The fault was attributed to the special Cummins engine installed by Python Mfg. to meet Caltrans emission requirements.

CARB Engine Communication Problems

In order to purchase the Python PHP machine, the diesel engine needed to meet mandatory California vehicle emission regulations. The only available approved engine at the time was an experimental Cummins CARB 2007 diesel engine. Python Mfg. only offered a John Deere Tier 3 diesel engine, the standard engine that the Python PHP control system was designed around. Python Mfg. agreed to modify their standard machine to accommodate the new Cummins engine for Caltrans' use in California. To accomplish this, Python Mfg. added an additional computer controller to enable the Cummins engine computer to communicate over the Controller Area Network (CAN) bus with the standard transmission computer and the patching machine control computer. The fault proved difficult to diagnose due to the intermittent occurrences only present at full power and a series of ambiguous error messages recorded by the computer. After many months, the cause of the problem was determined to result from a CAN bus overload error. Mitigating the issue proved to be difficult and time consuming. Ultimately, a local equipment service company proficient in CAN bus diagnostics, programming, and configuration resolved the communication errors. Slowing down the transmission computer data rate, adding appropriately sized CAN bus termination resistors, and making some minor control programming changes rectified the fault issue. Python Mfg. has since advanced their design such that future versions of their machine that utilize the Cummins engine will not include the supplementary conversion computer, which ultimately caused the problem.

Machine Software Upgrades

Python Mfg. released software upgrades to improve general machine functionality and performance. During the Python PHP machine's active service with Caltrans in San Jose, AHMCT Research Center personnel installed a software upgrade, and the machine immediately exhibited intermittent power loss problems. Rolling back the machine's program to the previous software version mitigated the problems. Since the Python PHP machine was operational and utilized by Caltrans San Jose crews at the time, the AHMCT Research Center postponed the software upgrades in favor of allowing the machine to continue highway patching operations with the original software.

After the Python PHP machine's removal from the D4 trials and return to UC-Davis, the AHMCT Research Center worked with Python Mfg. to solve the updated software issue. Since the AHMCT Research Center worked with Python Mfg. to develop additional machine capabilities, the updated software needed to be installed on the machine to continue research development work. However, installing the upgraded software re-created the Cummins engine communication issue. Apparently, the new software releases excluded the software patches developed to adapt the Python PHP machine platform to the Cummins engine since only one

Caltrans Python PHP machine exists. The software can be customized for the Cummins engine. However, this task remains incomplete due to the machine's current status of being removed from the D4 deployment trials.

Material Conveyor Belt Sticking Issue

The hot asphalt material is transported on a conveyor belt running through the telescoping arm from the heated hopper to the front patching head of the machine. The hot asphalt rolls off the end of the conveyor belt and falls into the pothole. Asphalt material that gets stuck to the conveyor belt, which occurs more frequently with smaller sized aggregate asphalts, gets pulled back into the machine. Asphalt then builds up on the conveyor belt and other surfaces, thus jamming critical moving parts and fouling internal components. Python Mfg.'s original mitigation method involved coating the conveyor belt with citrus cleaner before and periodically during its operation. Relying on citrus cleaner alone proved to be insufficient to prevent asphalt build-up and was an expensive solution due to the high cost of citrus cleaner. During highway operations in the D4 Foster City maintenance area, the asphalt build up on the conveyor belt and the patching head became so problematic that the operator would need to stop periodically on the highway and scrape the belt clean. Manually scraping the belt clean may be a reasonable practice for secondary roads but is unacceptable for mainline moving closure patching operations.

The AHMCT Research Center worked with Python Mfg. to develop a spiral brush attachment at the end of the conveyor (location a in Figure 10) that physically scrubs the belt clean to prevent sticking asphalt from building up on the belt or getting pulled back into the machine. The spiral brush is chain-driven off of the conveyer drive roller shaft and geared up such that the brush spins faster than the belt. The Python PHP machine was returned to UC-Davis for the brush modification, which also involved the modification of the compaction roller water spray heads (location b in Figure 10). The modified Python PHP machine was then returned to the D4 San Jose maintenance station for testing on the highway. The successful conveyor brush development completely prevented asphalt from building up on the material belt and the patching head.

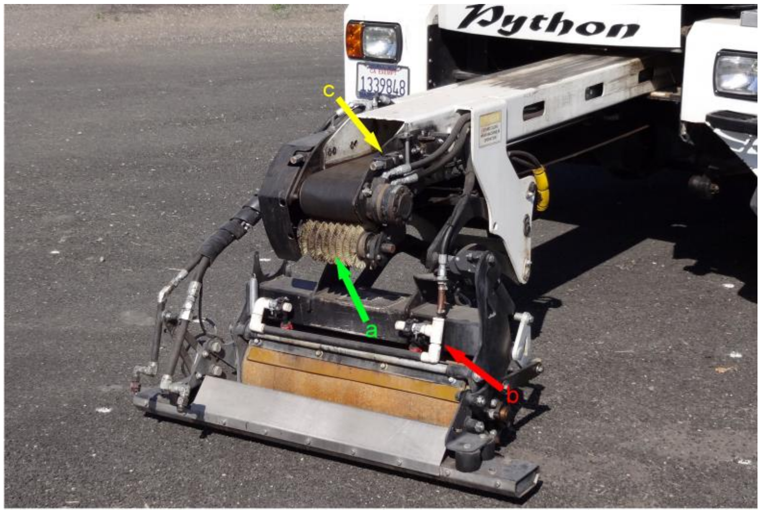


Figure 10. AHMCT Belt Tensioner

Conveyer Belt Tensioner Upgrade

Unfortunately, the installation of the conveyor brush assembly upgrade made access to the belt tension adjustment screws very difficult. Changing the conveyor belt or simply adjusting the conveyor belt tension became a long process of disassembling a significant portion of the spiral brush assembly to access the tension screws. AHMCT Research Center personnel modified the conveyor drive drum mounting to allow direct access to the conveyor belt tension screws from the front of the arm without any disassembly required (location c in Figure 10). This feature further simplified the process of changing the conveyor belt and especially improved the convenience when adjusting the belt tracking, which requires the belt to be moving.

Conveyor Drive Roller Change

As part of the spiral brush mechanism upgrade, Python Mfg. replaced the existing conveyor drive roller with a newer version consisting of a longer shaft on one side to drive the brush assembly. Unfortunately, Python Mfg. selected a smooth steel surface on the replacement part instead of bonding a rubber surface similar to the original part. The conveyor belt immediately experienced intermittent slipping on the drive roller. The conveyor belt tension was increased, but this only exacerbated a belt off-tracking problem. The Python Mfg. solution included welding parallel steel traction ribs on the surface of the drive roller. These strips proved to be attached too high on the drive roller’s surface and diminished the effectiveness of the spiral brush attachment. AHMCT Research Center personnel machined down the drive roller ribs’ height on a lathe and seemingly resolved the problem. Python Mfg. also provided the AHMCT

Research Center with a replacement drive roller with the bonded rubber surface, should the Python PHP machine require a replacement drive roller in the future.

Conveyor Belt Tracking

Several factors can cause the material delivery conveyor belt to ride off the center track and rub on the sides of the arm tube, thus damaging and potentially jamming the belt. Python Mfg. manufactured this version of the patching machine with stationary conveyor belt guides attached at the base of the arm under the machine. These guides center the belt as it rides onto the idler roller and keep the belt tracking down the center of the telescoping arm tube. Whether the guides ever functioned correctly or if the addition of the spiral brush exacerbated their ineffectiveness remains unclear; in either case, the guides were deficient.

The AHMCT Research Center worked with Python Mfg. to replace the stationary guides with idler roller crowning and flanges. D4 maintenance and the AHMCT Research Center coordinated the transport of the Python PHP machine to UC-Davis, where AHMCT Research Center personnel installed the modified conveyor parts along with the improved conveyor drive roller discussed previously. To account for the new idler roller design, the AHMCT Research Center research team installed a new longer conveyor belt. Next, the Python PHP machine was returned to the Caltrans D4 San Jose maintenance yard and immediately reactivated into service. The modified idler roller design performed without issue during the highway patching trials.

Citrus Cleaner Incompatibility

The standard Python PHP cleaner, and the most commonly utilized asphalt cleaning fluid, is diesel fuel. Caltrans favors the use of an environmentally safe alternative fluid for asphalt clean-up operations. The AHMCT Research Center purchased the Python PHP machine modified to operate with concentrated citrus solvent instead of diesel fuel, as shown in Figure 11. Unfortunately, Python Mfg. used their standard diesel seals and hose materials, which dissolved over a short period of time. AHMCT Research Center personnel identified new materials that are chemically resistant to the citrus cleaner and replaced the machine’s cleaning system hoses and an all Viton seals fluid pump. The level of difficulty involved in changing the rubber hose in the spring return mechanical reel led to the decision to replace the hose reel at regularly scheduled intervals, as needed.

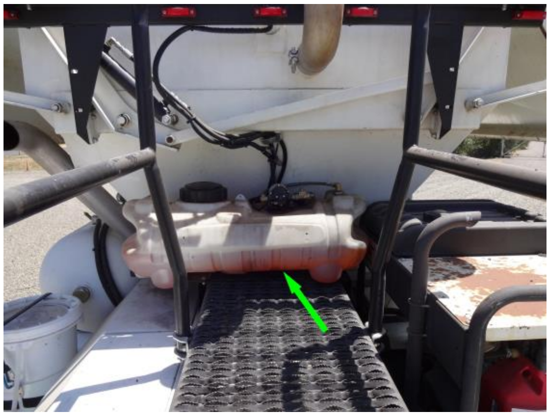


Figure 11. Citrus Cleaner System

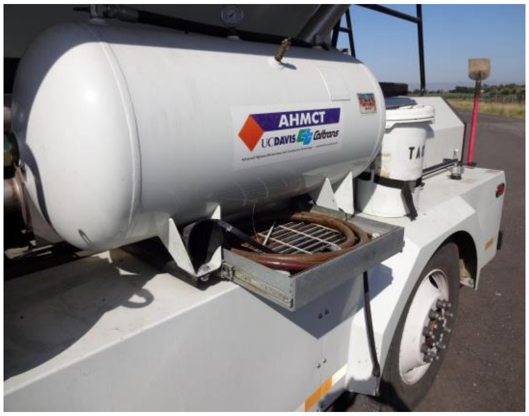


Figure 12. Python PHP Fill Hose Tray and Bucket

Tack Oil Utility Tray and Bucket

Coating a pothole with tack oil emulsion prior to filling promotes patch adhesion. This tack oil coating denotes the customary second step in the traditional hot asphalt pothole patching method. The Python PHP machine contains a tack oil pump designed to suck the tack oil from a bucket through the fill hose to fill a heated storage tank. Then for patching operations, a selector valve is switched, the fill hose is removed, and the pump supplies the tack oil to the spray nozzle on the patching head. At the end of patching operations, the tack oil is pumped back through the fill hose into the tack oil storage bucket. Also, a common practice involves sucking a mixture of citrus solvent and water through the fill hose to refill the tack oil reservoir, thereby reducing the possibility of clogging.

During the Python PHP machine D4 trial operations, the AHMCT Research Center discovered that Caltrans preferred purchasing just enough tack oil for a day’s operation while at the batch plant, as opposed to trying to store a bulk of tack oil in the yard. Therefore, the machine needed to carry both the fill hose and tack oil bucket on the machine. Python Mfg. did not make any allowance on the machine for the storage of the messy fill hose or the tack oil bucket. AHMCT Research Center personnel designed and built a drip tray drawer to contain the fill hose and a mounting for the tack oil bucket (Figure 12). Ultimately, the Caltrans D4 patching crews elected to skip the tack oil spray step altogether, which didn’t seem to cause any adverse effect on the long-term adhesion of patches. The elimination of the tack oil simplified the patching operation procedures and further reduced patching times—critical aspects for moving closure highway patching operations.

Redundant Safety Braking Circuit

The Python PHP machine, a hydrostatic drive vehicle, necessitates advanced drive train computer control. The Python PHP machine contains both a conventional hydraulic braking system in the wheels and a fail-safe parking brake module mounted in-line in the driveline. The driver actuates the hydraulic brake by directly pressing on a foot-activated brake pedal inside the cab. Naturally, the driver lets off the accelerator pedal when applying the brake because accelerating during braking substantially reduces braking efficiency. In this type of hydrostatic-drive vehicle, a computer controls the engine's speed. As such, the accelerator pedal simply becomes a drive train computer input signal. This arrangement raises the possibility that a drive train computer malfunction could result in the vehicle accelerating as the driver attempts to slow down the vehicle, diminishing control and braking capabilities. Therefore, Python Mfg. added an electric override to assist with hydraulic braking, should acceleration and braking occur simultaneously.

During the Python PHP machine D4 trial patching operations, the operator noticed a drag on the vehicle that caused the brake rotors to become extremely hot while driving on the highway. Notified of this issue, Python Mfg. believed a failure within the driveline hydraulic emergency brake module mounted on the transmission triggered the problem. Python Mfg. arranged to have the component replaced in West Sacramento, but the problem persisted. Subsequently, the problem was correctly diagnosed as an anomaly in the electrical override braking circuit and successfully mitigated.

Compaction Pressure Limit

A hydraulic cylinder and linkage mounted under the end of the telescoping arm actuates the compaction roller on the Python PHP machine patching head. Between the cylinder force and the mechanical linkage advantage, the compaction roller down pressure can almost lift the front of the machine off the ground. Therefore, knowing when to stop driving down the patching head in order to limit down pressure on the compaction roller requires a fair amount of operator skill and experience. In practice, the operator frequently made several adjustments to the compaction pressure during the patching of each pothole, which added appreciably to overall patch application times. Since conducting patching operations in moving closures requires very quick patch times, a solution to automate the application of compaction pressure became a necessity.

The AHMCT Research Center worked with Python Mfg. to reprogram the machine control program enabling regulation of the compaction down pressure. As the compaction roller presses against the pavement, the associated hydraulic circuit pressure increases proportionally, generating the compaction pressure. The Python PHP computer reads this pressure and disregards the down command switch once the down pressure limit is reached. The operator can set the desired roller compaction pressure on the in-cab control panel, thus producing a more consistent patch. This feature accelerates and enhances the patching process by enabling the operator to drive the patching head down, make firm contact with the pavement, release the switch, and ensure a consistent compaction pressure.

Asphalt Hopper Step

Python Mfg. provides a slide-out stairway up the rear of their machine for the operator to gain access to the hopper, which is mostly useful for soaping the hopper walls prior to loading asphalt and for post-operational cleaning. However, sometimes during highway patching operations, the operator may want to quickly look in the hopper to examine the state of the remaining hot asphalt. The rear step is not readily accessible on the highway in moving closure operations. AHMCT Research Center personnel added a step adjacent to the cab access door, such that the operator can step out the door and onto the gangway and then step up on the footstep to safely look into the hopper while out on the highway (Figure 13).

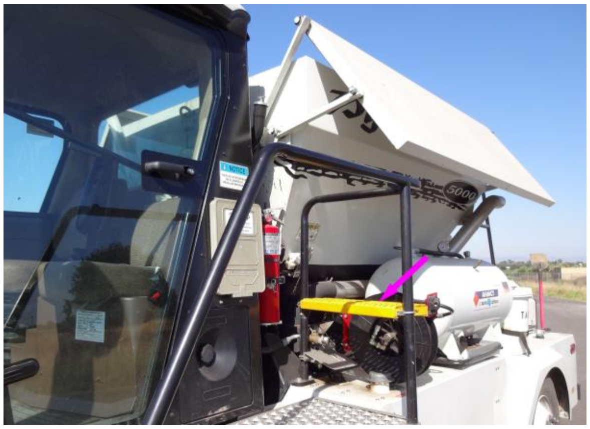


Figure 13. Hopper Viewing Footstep

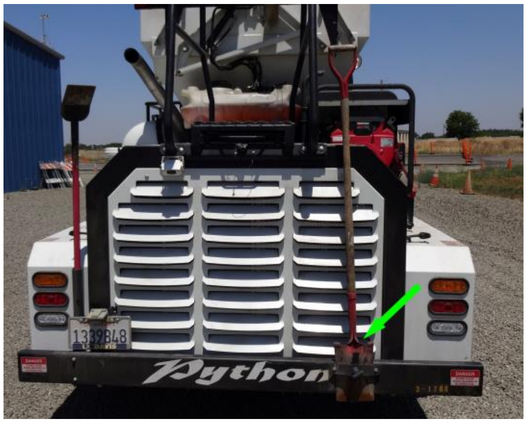


Figure 14. Hopper Shovel

Hopper Asphalt Shovel

The Python PHP machine came standard with a rear bumper mounting for a pole scraper, a useful tool for scrapping asphalt from the conveyor belt and chipping asphalt from the patching head. The pole scraper's lack of reach limits this tool's usefulness in the hopper. The AHMCT Research Center fabricated a long-reach shovel scraper, which enables the operator to reach into the hopper from the rear gangway and chip loose any remaining clumps of stuck asphalt so it falls into the auger during the machine's post-operational cleaning (Figure 14). The shovel scraper contains a perpendicular end grip to maximize the power and reach of the shovel. AHMCT Research Center personnel designed a spring-clamp mounting, which stores the shovel scraper securely on the Python PHP rear bumper and can be released with the pull of a single T-pin.

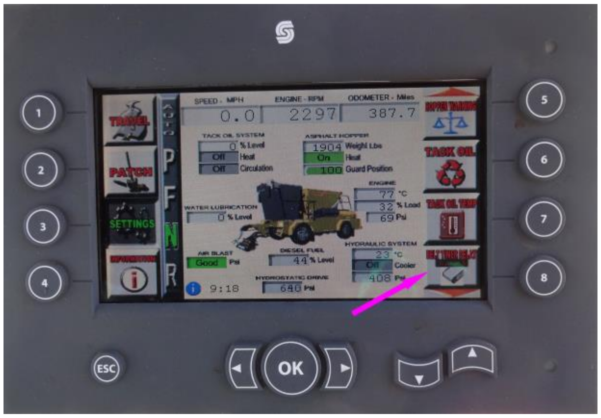
Reverse Drive Power Loss

Near the end of the Python PHP machine trial deployment period, the machine lost all power during reverse driving. A faulty electrical connector on the reverse drive hydraulic solenoid valve contributed to the power loss issue. Replacing the connector resolved the problem.

Conveyor Belt Extended Control

Heaters keep the asphalt in the hopper hot and ready for patching. As the asphalt leaves the hopper and is conveyed to the patching head on the material supply belt, the patching material no longer receives heat and promptly cools down. Therefore, Python Mfg.'s standard control protocol includes running the conveyor a few seconds after the operator releases the material delivery button to clear the remaining hot asphalt off the belt before it cools down. Machine operators must learn how to account for the quantity of asphalt left on the conveyor belt and release the material feed button in advance to prevent overfilling the pothole. Upon initially pressing the button, the nine-foot asphalt conveyor results in a six- to ten-second delay before the asphalt begins to shed the patching material off of the conveyor belt and into the pothole. Likewise, a nearly equal stop delay keeps the belt running upon the release of the asphalt feed button. Added together, these delays significantly lengthen the overall machine pothole patch times. Figure 15 shows the "Belt Delay" screen in the settings menu.

To attain the quicker patch application times necessary to support mainline highway moving closure operations, the AHMCT Research Center worked with Python Mfg. to develop greater conveyor control capabilities that provided the operator access to the material supply belt stop delay time on the control panel. This feature enables the operator to suspend the belt-clearing feature and leave asphalt on the conveyor between potholes (Figure 16). This enhancement eliminates the initial material delivery delay and the asphalt stop delay time to shave roughly twenty seconds off the Python PHP machine’s patch application time. This practice proves especially effective with potholes spaced minutes apart on the highway such that the asphalt on the conveyor remains appreciably heated. For longer distances of travel, the operator can choose to clear the asphalt off the conveyor in a pothole or off the side of the highway away from traffic.


SPEED - MPH 0.0
ENGINE - RPM 2297
ODOMETER - Miles 387.7

TACK OIL SYSTEM
0 % Level
Off Heat
Off Circulation

ASPHALT HOPPER
1904 Weight Lbs
On Heat
100 Guard Position

ENGINE
77 °C
32 % Load
69 Psi

WATER LUBRICATION
0 % Level

AIR BLAST
Good Psi

DIESEL FUEL
44 % Level

HYDRAULIC SYSTEM
23 °C
Off Cooler
408 Psi

HYDROSTATIC DRIVE
640 Psi

9:18

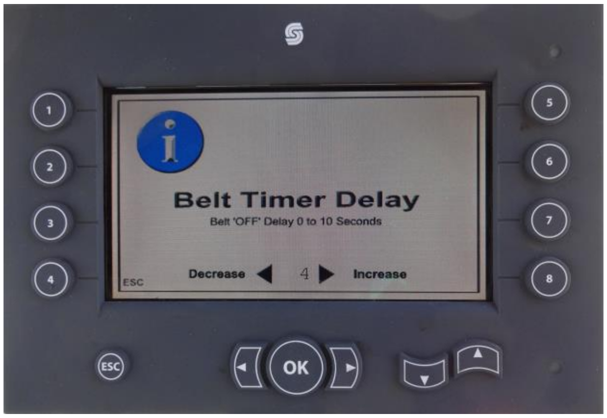

Belt Timer Delay
Belt 'OFF' Delay 0 to 10 Seconds
Decrease ◀ 4 ▶ Increase
ESC

Figure 15. Settings Screen – Belt Delay

Figure 16. Belt Delay Adjustment

Automated Guard Control

An auger screw drive in the bottom of the Python PHP V-shaped hopper feeds hot asphalt forward toward a chute, which discharges over the material conveyor belt. When the operator presses the asphalt feed button on the command joystick, the auger rotates and asphalt drops onto the conveyer belt and then moves forward to the end roller, where the asphalt falls off into the pothole (location b in Figure 17). The auger is incapable of spinning with the weight of a full load of sticky asphalt pressing against the auger in the hopper. The Python PHP design relies on a retractable guard riding along a support pipe to partially cover the auger and thereby reduce the dead weight pressing against the auger (location a in Figure 17). On the leading edge of the guard resides a set of three fins (Figure 18), which breaks up asphalt chunks or frees the auger should it jam by hydraulically driving the guard back and forth over the auger. The Python PHP machine’s standard operating procedure permits the operator to periodically retract the guard as the asphalt discharges from the hopper, ensuring at least a couple of feet of auger stays covered with hot asphalt. In practice, however, the operator retracts the guard only after noticing the asphalt beginning to thin out unexpectedly or stop coming off the conveyor altogether.

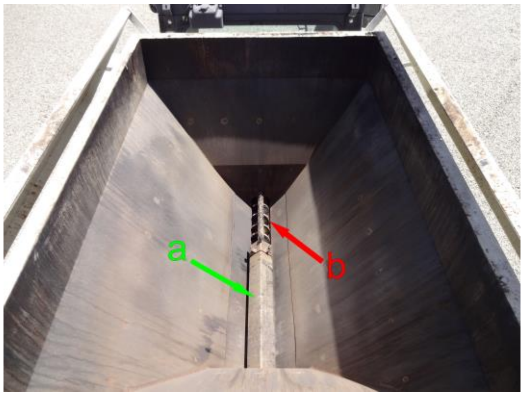


Figure 17. Python PHP Hopper with Guard Closed

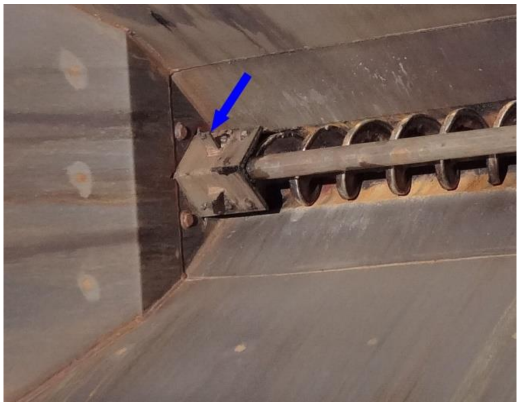
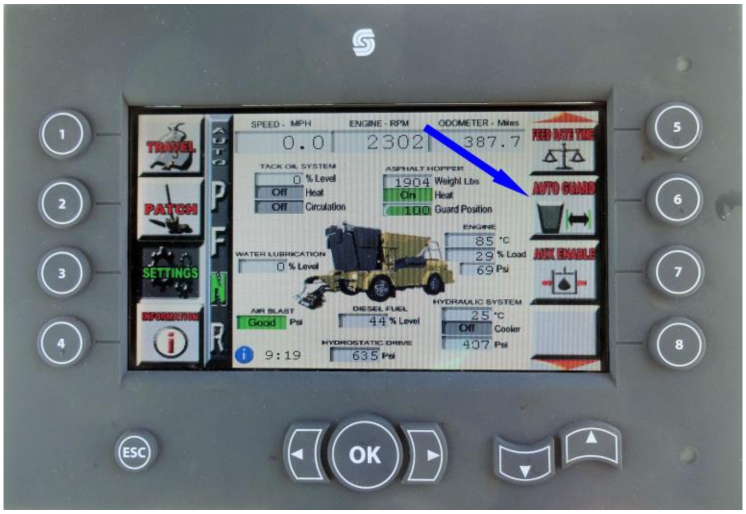


Figure 18. Python PHP Guard Fins

The quick delivery of hot asphalt material to potholes without any delays remains an essential aspect for Caltrans mainline moving closure patching operations. The operator does not have time to access the guard settings when stopped at a pothole or while driving along the highway between potholes. Therefore, the AHMCT Research Center collaborated with Python Mfg. to add an automated guard control function to the control program. Based on the weight of the asphalt in the hopper, the controller opens the guard as the asphalt depletes from the hopper. The Auto Guard program settings can be accessed through the **SETTINGS** tab on the control panel (Figure 19). Selecting the **AUTO GUARD** button displays the auto guard adjustment screen, where the desired end positions of the guard can be entered (Figure 20). The initial set-up process required the use of this adjustment; however, the operator unlikely would ever need to access this setting under normal operating conditions.


SPEED - MPH 0.0
ENGINE - RPM 2302
ODOMETER - Miles 387.7

TACK OIL SYSTEM
0 % Level
Off Heat
Off Circulation

ASPHALT HOPPER
1904 Weight Lbs
On Heat
100 Guard Position

ENGINE
85 °C
29 % Load
69 Psi

WATER LUBRICATION
0 % Level

AIR BLAST
Good Psi

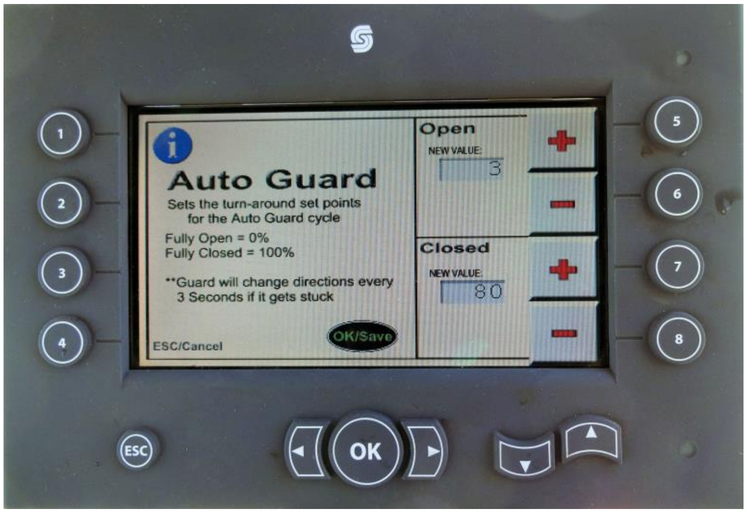
DIESEL FUEL
44 % Level

HYDRAULIC SYSTEM
25 °C
Off Cooler
407 Psi

HYDROSTATIC DRIVE
635 Psi

9:19

Figure 19. Settings Screen - Auto Guard


Auto Guard

Sets the turn-around set points for the Auto Guard cycle

Fully Open = 0%
Fully Closed = 100%

**Guard will change directions every 3 Seconds if it gets stuck

Open
NEW VALUE: 3

Closed
NEW VALUE: 80

Figure 20. Auto Guard Adjustment

Expanding Use of the Python PHP Machine

Safety and production benefits were demonstrated during the deployment of the Python PHP machine. For the first time, the development of moving closure patching operations efficiently extended the functionality of this equipment to the mainline highway, where Caltrans' most critical need exists. However, Caltrans must consider fleet reduction mandates before considering implementation of these machines. For a successful integration of any semi-automated pothole patching machine into a Caltrans maintenance operation, the equipment must demonstrate the potential to log significant operational hours or be indispensable to the Caltrans maintenance mandate. Caltrans maintenance can continue with conventional manual patching methods without access to a semi-automated patching machine, so the indispensability justification is not present. In addition, a high enough concentration of mainline highway potholes to log the necessary operational hours and justify retaining the machine likely does not exist in a single maintenance area. A potential solution involves operating the machine over multiple maintenance areas, the strategy utilized in the Caltrans District 4 Python PHP machine deployment trial. Expanding capabilities, functionalities, and utilities to semi-automated patching equipment yields another potential solution to increase semi-automated patching machine usage.

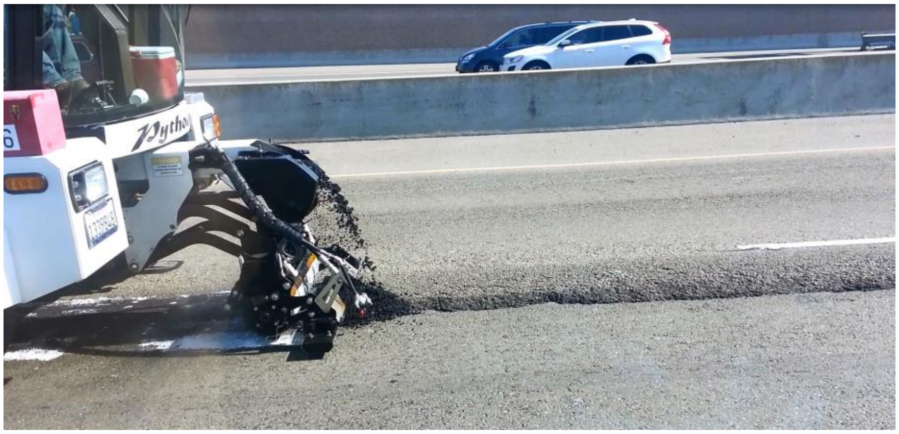


Figure 21. Linear Pavement Gap Patching

Linear Patching Option

By definition, potholes primarily represent a single, finite hole, as opposed to the linear, unbounded gaps often found along longitudinal asphalt concrete/ Portland Cement Concrete (AC)/ (PCC) transitions and ramp transitions. Semi-automated pothole patching application heads are designed to apply finite area patches. However, patching an unbounded linear pavement gap can be accomplished by applying a series of spot patches as the machine moves along the linear gap. This patching approach is feasible for short segments, but it quickly becomes infeasible in a moving lane closure as the length of a linear pavement gap stretches past three or four patches (Figure 21). The strategy towards solving this problem involved developing a continuous, linear hot asphalt patching process, which would improve the quality and efficiency of linear gap patching. The new capability would serve as an add-on to the existing patching capability and should be easily selectable by the operator in the cab.

The AHMCT Research Center developed the necessary characteristics of a continuous patching mode feature and worked with Python Mfg. to modify the Python PHP machine to support this new functionality. The hardware aspect of the machine modification involved the addition of a funnel attachment to form-up the asphalt into a neat linear mound on the pavement, which is then compacted into the pavement gap by the trailing compaction roller (Figure 22). Thick rubber strips bolted along the bottom of the funnel's wings at an appropriate length create the desired height of hot asphalt mound. This hardware is now standard on all models of Python PHP patching equipment. Further improvements included a lateral drive mechanism to control the spacing of the wings from in the cab and the replacement of the flat rake with a comb-style rake (Figure 23). Both of these modifications greatly benefitted patch quality by enabling the operator to effectively move and evenly distribute the asphalt around the pothole.

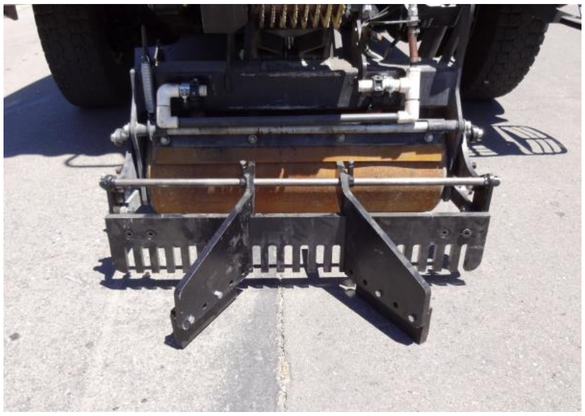


Figure 22. Python PHP Rake & Funnel Attachment

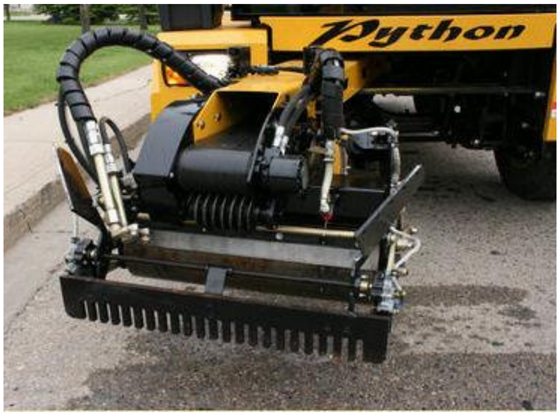
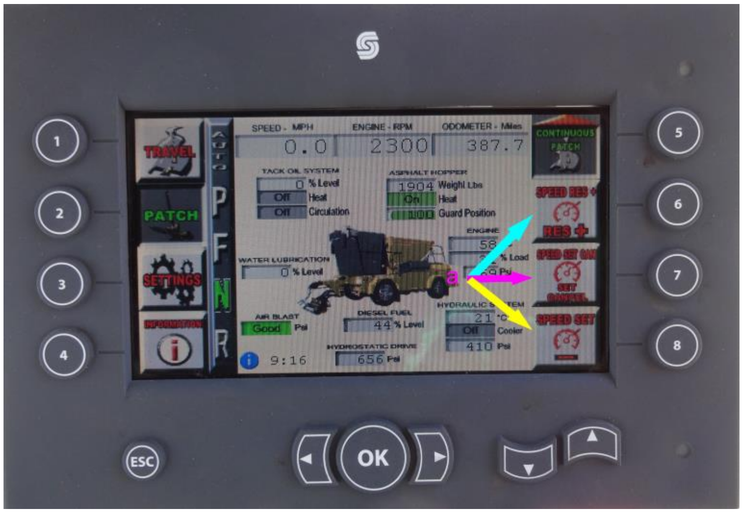


Figure 23. Python PHP Updated Patching Head

The addition of continuous patching capability required minor modifications to the Python PHP machine control program. AHMCT Research Center personnel added a **CONTINUOUS PATCH** mode selection on the control panel, which configures the Python PHP machine for continuous patching operation (Figure 24). Upon selection, a second screen appears showing the operator how the machine controls transform to accommodate continuous patching mode operation (Figure 25). The operator can adjust two interdependent machine settings on the control panel prior to running in continuous patching mode to produce the desired patch. The asphalt delivery rate can be adjusted in the control **SETTINGS** screen (Figure 26) and relates to both the machine speed and the size of the pavement gap. The machine speed setting essentially acts as a low-speed cruise control that drives the Python PHP machine at a continuous but adjustable speed, which is set for each specific application. The operator drives the Python PHP machine at the desired speed and presses the **SPEED SET** button, which holds that speed. The operator can then make fine speed adjustments by pressing the **SPEED RES +** button or cancel the setting with the **SPEED SET CAN/ SET CANCEL** button (location a in Figure 24).



SPEED - MPH 0.0
ENGINE - RPM 2300
ODOMETER - Miles 387.7

TACK OIL SYSTEM
0 % Level
Off Heat
Off Circulation

ASPHALT HOPPER
1904 Weight Lbs
On Heat
100 Guard Position

ENGINE
58
% Load
Psi

WATER LUBRICATION
0 % Level

AIR BLAST
Good Psi

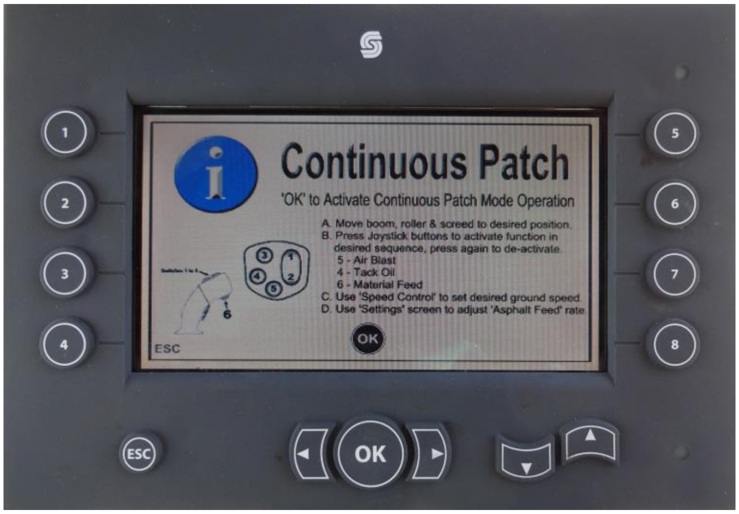
DIESEL FUEL
44 % Level

HYDRAULIC SYSTEM
21 °C
Off Cooler
410 Psi

HYDROSTATIC DRIVE
656 Psi

9:16

Figure 24. Continuous Mode



Continuous Patch

'OK' to Activate Continuous Patch Mode Operation

A. Move boom, roller & screed to desired position.
B. Press Joystick buttons to activate function in desired sequence, press again to de-activate.
5 - Air Blast
4 - Tack Oil
6 - Material Feed
C. Use 'Speed Control' to set desired ground speed.
D. Use 'Settings' screen to adjust 'Asphalt Feed' rate.

Figure 25. Continuous Mode Controls

Once the operator configures the continuous mode, he or she can quickly switch the Python PHP machine between the normal pothole patching mode and the continuous mode as needed in the moving closure operation. Switching into continuous mode on the Python PHP machine changes the buttons on the joystick from momentary to on/off toggle controls for the air blast, tack oil, and material feed functions. The operator initiates the continuous patching process, which the machine automatically continues. With the machine operating in continuous mode, the operator only needs to steer the machine, remaining centered on the linear pavement gap. The AHMCT Research Center verified that the continuous patch functionality on the Python PHP machine correctly functioned in trials at UC-Davis, but it has yet to be utilized on the mainline highway in a Caltrans moving closure patching operation.

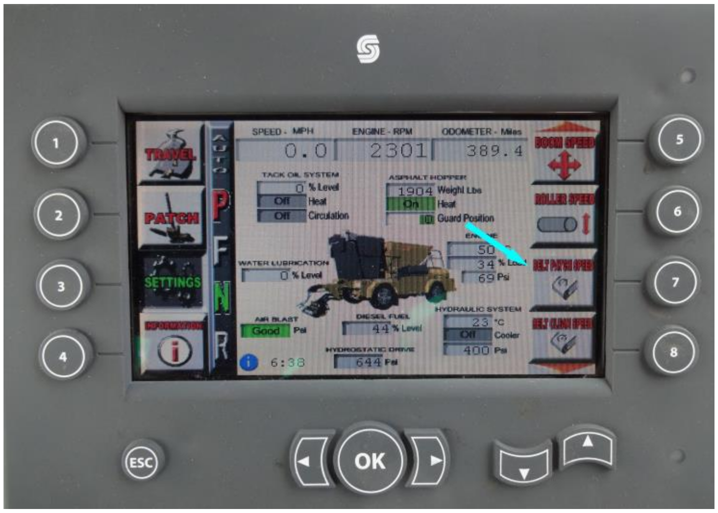


Figure 26. Asphalt Delivery Rate Setting

Edge Drain Patching Option

During deployment testing with Caltrans, maintenance personnel identified an innovative application of the continuous patching function. In the 1990s, Caltrans retrofitted many miles of highway shoulder edge drains to reduce damage occurring in existing PCC pavements [5]. The process involved cutting a four-inch wide trench in the asphalt shoulder along the PCC edge, installing a drainage pipe, backfilling with aggregate, and then patching with asphalt. These retrofits are extremely susceptible to block cracking and are difficult to crack seal, leading to quick deterioration of the retrofit edge drain pavement (Figure 27). The block cracking eventually loosens, allowing chunks to be dislodged by traffic impacts until the retrofit edge drain pavement resembles a linear series of small potholes. Caltrans maintenance expressed strong interest in utilizing the Python PHP machine to patch these failures. Adapting the Python PHP machine's continuous patching mode to patch these failures would be especially beneficial when pavement deterioration reaches an advanced stage; e.g., when there is a significant amount of dislodged pavement blocks.

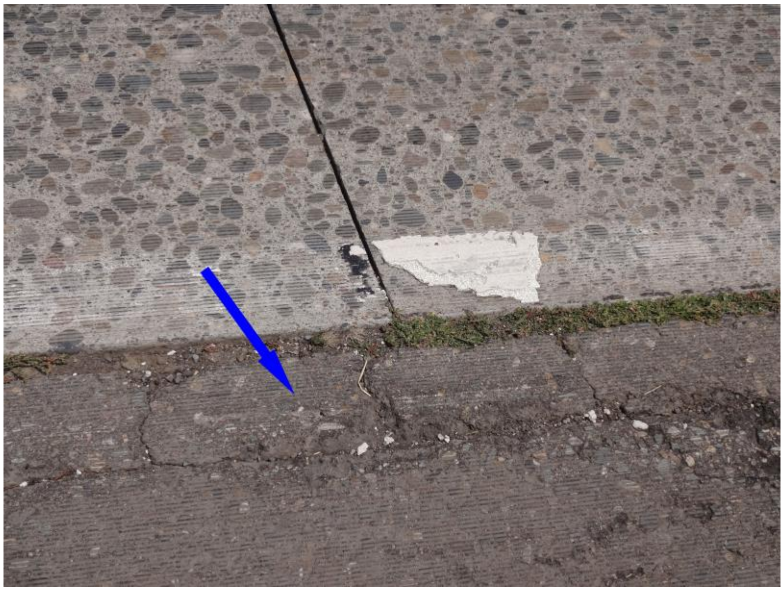


Figure 27. Edge Drain Pavement Failure

Whether the operator continuously patches edge drains or continuously patches linear pavement gaps, the Python PHP machine's functionality and control remain the same. However, an additional benefit may exist for edge drain patching, which boosts efficiency. Since edge drain gaps mostly exist on the asphalt shoulder, the Python PHP machine can potentially access and patch the edge drain while operating from the shoulder. In addition, edge drains crossing highway traffic ramps pose the greatest traffic impact, and these edge drain failures are the greatest concern, since they are essentially crossing a traffic lane. The Python PHP machine can transition from the shoulder to patch across traffic ramps in a moving closure operation and then transition back onto the shoulder. This type of quick moving ramp closure, which typically utilizes two MAZEEP vehicles to block ramps, remains a common practice for Caltrans maintenance.

Since the wings on the Python PHP funnel attachment are driven laterally and are equidistant from the center of the patching head, the Python PHP machine must be driven along the center of the edge drain. When operating the Python PHP machine from the shoulder, this configuration places the Python PHP machine approximately four feet into the traffic lane, which necessitates the closing of a lane. AHMCT Research Center personnel built a special edge drain attachment for the Python PHP machine, which enables the head to fully rotate and patch out near the side of the machine. Being at full rotation, however, positions the patching head at a fifteen-degree angle relative to the vehicle centerline. The angled edge drain attachment accounts for this offset (Figure 28). Utilizing the edge drain attachment limits the machine's encroachment into the traffic lane to approximately two feet (Figure 29). The edge drain attachment simply bolts onto the flat rake and can be flipped for either left or right side patching operations. Similar to the Python PHP funnel attachment wings, thick rubber edge strips form the sides of the funnel, and their width dictates the asphalt mound height.

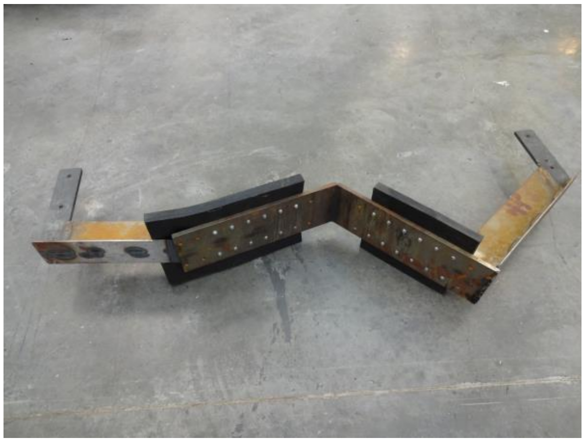


Figure 28. Edge Drain Attachment

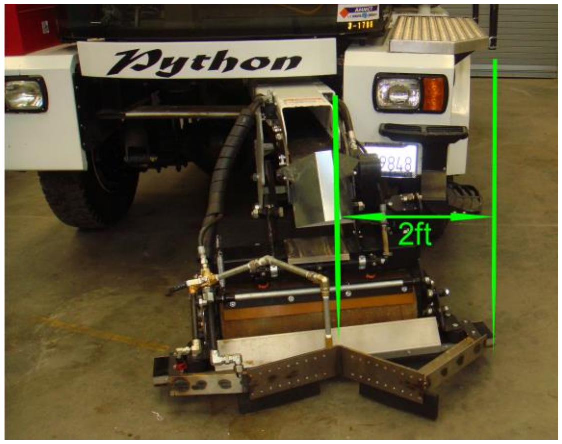


Figure 29. Edge Drain Patching Configuration

Removing the cracked block asphalt in the failed edge drain prior to patching could potentially enhance the effectiveness of edge drain patching. The jagged edges of the asphalt blocks prevent the loosened asphalt chunks from being easily removed. The Caltrans Division of Maintenance could potentially benefit from a complimentary machine capable of efficiently removing failed pavement from the retrofit edge drains. Ideally, this machine would also collect all removed pavement pieces and dust, thus leaving the edge drain gap clean and ready for patching potentially in a single pass. An appropriate vacuum would be necessary to meet air quality regulations in most metropolitan areas of the state. In order to compliment the Python PHP machine, the machine would ideally be a street-legal vehicle capable of operating in moving highway closures or from the shoulder. This would necessitate the machine's ability to be remotely operated from inside the vehicle, such that any workers would not be on foot and exposed to direct traffic on the highway. In addition, the machine should not extend more than two feet into the traffic lane from the shoulder in operation. The AHMCT Research Center conducted a search of equipment and was unable to identify any commercially available machines or suitable equipment that could potentially be modified to meet these requirements. Accomplishing this specific task apparently would require machine design and development.

Asphalt cutting or routing, which far surpasses all other types of passive cleaning methods like wire brushing and compressed air either separately or in combination, remains the most consistent and cost-effective method of clearing pavement gaps. This clearing technique especially applies in the case of patching edge drains where the relatively large asphalt chunks must be removed in order to enable the new asphalt patch to properly adhere to the intact pavement sidewalls. Commercially available pavement cutters like the Bobcat Wheel Saw attachment (Figure 30) would be a suitable cutter in the development of a high production routing machine.

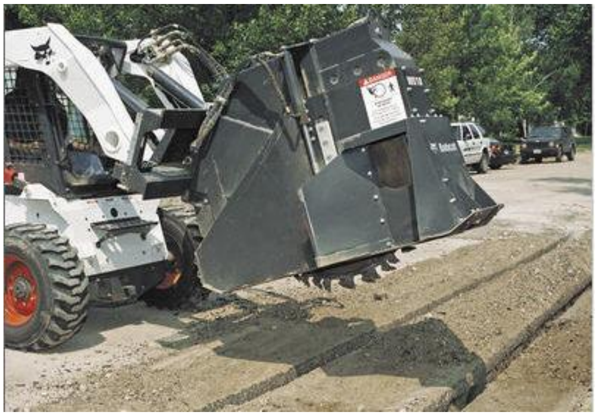


Figure 30. Bobcat Wheel Saw Attachment

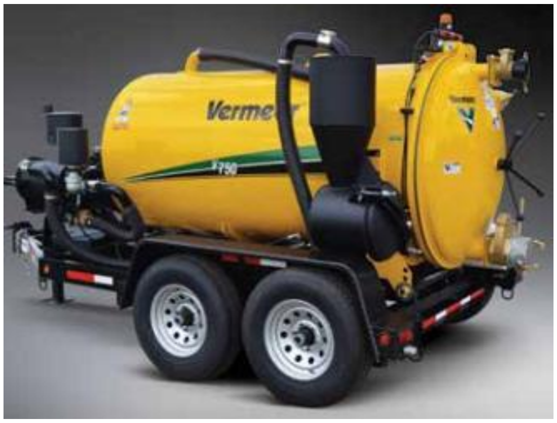


Figure 31. Pavement Vacuum

When conducting mainline highway shoulder routing operations, the large amount of routed pavement debris and dust ejected in the process must be collected to avoid harm to the high-speed traffic passing by and for environmental reasons. The lighter materials can be collected with a large vacuum unit with a cyclone dust collector (Figure 31). A secondary conveyor may be necessary to collect any large asphalt chunks dislodged by the cutter. A small amount of water may be injected into the air stream to help reduce dust in the vacuum exhaust, but the asphalt tailings must remain dry. The cutter-saw head vacuum debris collection system, together with a high-pressure and flow rotary air compressor, could be mounted on a mid-sized truck chassis. The asphalt router cutter head mechanism should be configured to operate off of either side of the vehicle and remotely steered from inside the cab. In addition, incorporating a secondary tracking wheel, which could autonomously track the longitudinal gaps and translate freely upon the release of the cutting head, would add convenience to this routing process.

The design and development of a routing machine would support the Python PHP edge drain patching operation and could support the Sealzall crack sealing machine that was also developed at the AHMCT Research Center for use in Caltrans operations. To obtain this dual functionality, the width of the asphalt cutter would need to be adjustable.

Permanent Spall Repairs of Rigid Pavements

Temporary Asphalt Spall Repairs

Caltrans recommends against the use of bituminous materials for permanent repairs of rigid pavements because those materials allow excessive horizontal movement of adjacent slabs, provide no load transfer across transverse joints, and may lead to very rapid asphalt patch deterioration. Bituminous materials should be only considered as a short-term or temporary repair [6]. Since a permanent spall repair involves a full work crew and a fixed lane closure, Caltrans maintenance crews routinely apply ad hoc cold asphalt patches to mainline spalls as a short-term traffic mitigation technique until the resources can be attained to fix the spall permanently. The Python PHP machine applies a semi-permanent, compacted hot asphalt patch, which may prove to be superior to temporary cold asphalt patches based on limited mainline highway trials. The Python PHP machine’s compacted hot asphalt spall patches can be applied quicker than cold patches and may remain intact for many years. However, since these patches consist of bituminous materials, the patches cannot be classified as permanent and the spalls require a permanent repair at a later date. In addition, patching light colored PCC spalls with black asphalt can confuse drivers; traveling motorists may mistake the black asphalt patch as a pothole and swerve to avoid driving over it. However, urethane patches, which range from light gray to yellow in color, allow blending with the light beige concrete surface to create a more uniform pavement appearance.

Permanent Epoxy Gravel Spall Repairs

In practice, the application of permanent spall patches is done only when absolutely necessary on the highway. All of the prevailing methods of applying permanent spall patches in highway pavement are manual labor operations. Therefore, Caltrans maintenance crews establish fixed temporary lane closures to make these repairs on mainline PCC highways. Due to the widely dispersed nature of spalls, usually one or two high-priority spalls can be contained in a stationary lane closure. Furthermore, since establishing a stationary lane closure on the highway mainline normally requires a crew of five workers, a full crew must be dispatched for each repair, even if applying the actual patch requires fewer workers. This typical practice represents a significant time investment to patch a relatively small hole in the pavement. In addition, the lane closures must take place within a strict time window often dictated by the Caltrans Traffic Operations unit. This time constraint limits Caltrans crews in most metropolitan areas to placing only a single mainline stationary lane closure per shift. For these reasons, the access to the spall is a costly aspect of the concrete spall permanent repair process.

Once Caltrans makes the investment in establishing a mainline stationary lane closure to repair a spall, the focus is on creating a long-lasting patch by taking the time necessary to make a quality repair. Therefore, Caltrans crews are instructed to remove all damaged concrete in the repair area and cut or chip vertical faces and square corners in the pavement around the spall to guarantee a maximum strength repair. Removing all loose dust and debris helps ensure a clean and dry spall before applying the patch to promote optimum adhesion. Then, coating the spall with an epoxy resin material before adding pea gravel (i.e., the incompressible filler material) ensures consistency. Next, adding additional epoxy resin such that the gravel stays completely immersed in the epoxy and brought up to grade completes the patch application. Generally dispensed with a 20-minute set time, the two-part epoxy gravel repair patch should not be subject to traffic until the epoxy fully hardens (Figure 32). A two-part epoxy gravel patch installed within these guidelines will generally remain intact indefinitely while exposed to mainline highway traffic.

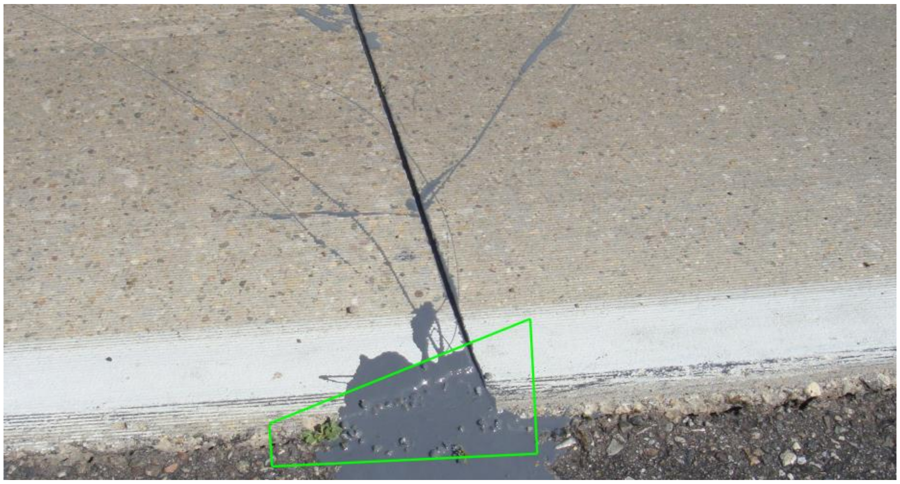


Figure 32. Caltrans Urethane and Gravel Spall Patch – Interstate 8

Python PHP Urethane Gravel Patching Benefits

The prospect of utilizing the Python PHP platform to apply permanent epoxy gravel spall patches on mainline highways would eliminate the fixed lane closure requirement and reduce the overall permanent spall repair time from hours to a minute or less. Moving lane closure spall patching would also significantly reduce the obstruction of highway traffic and provide a significant safety benefit by eliminating all workers' direct traffic exposure. Caltrans achieved these same safety benefits when utilizing the Python PHP machine in mainline, moving closure, hot asphalt, pothole patching operations and could experience these benefits with the added application of a permanent urethane patch, as needed.

Configuring the Python PHP machine to dispense quick set two-part epoxy materials also provides additional material handling and management benefits for Caltrans maintenance operations. Planning a hot asphalt patching operation with the Python PHP machine requires scheduling with a nearby asphalt batch plant to ensure the availability of the asphalt, purchasing a one-ton minimum, dumping any unused material, and cleaning the machine rigorously in accordance with a post-operational regimen. The two-part epoxy can be stored long-term in the machine and be used anytime with short notice. Stored and pumped separately, the epoxy resin and catalyst only come in contact in the disposable mixing nozzle at the point of application, initiating the chemical hardening process. When configured for epoxy patching, the Python PHP machine can be used to patch mainline spalls in a moving highway closure on short notice. The operator only needs to twist on a fresh mixing nozzle to begin patching. Dispensing epoxies with the Python PHP machine makes patching a small number of spalls, or even a single spall, far more practical than the Python PHP hot asphalt patching operation.

Ultra-Fast Urethane Set Times

Caltrans standards endorse a wide variety of permanent, partial depth PCC repairs and suggest that environmental factors such as placing and cure time can be key aspects in selecting the most suitable patching material for a specific site [4]. Polyurethane epoxy materials possess the fastest set (time to partial working strength) and cure (time to full strength) times, which is a determining factor when conducting moving closure patching operations. Caltrans successfully used this two-part urethane material for many years in spall repair and in slab crack/joint sealing. To support the proposed Python PHP mainline moving closure patching operations, the ratio of the catalyst would be increased to bring the urethane set-time down to nearly one minute. The goal includes matching the urethane set-time with the time of the Caltrans moving lane closure such that the newly placed patch supports the trailing highway traffic. The Python PHP machine would straddle the fresh patch, and the trailing Caltrans traffic control would also receive instructions to avoid driving on the fresh patch in order to maximize the protected set-time, which could exceed one minute. The urethane set-time cannot realistically be set below one minute due to the potential for clogging the mixing nozzle. Once the two components interact in the mixing nozzle, the hardening reaction commences. Consequently, the standard procedure involves intermittently dispensing a small amount of urethane through the nozzle to keep the mixing nozzle from clogging. For the Python PHP application, this nozzle refreshing would be automated and dispensed into a disposable container mounted to the patching head.

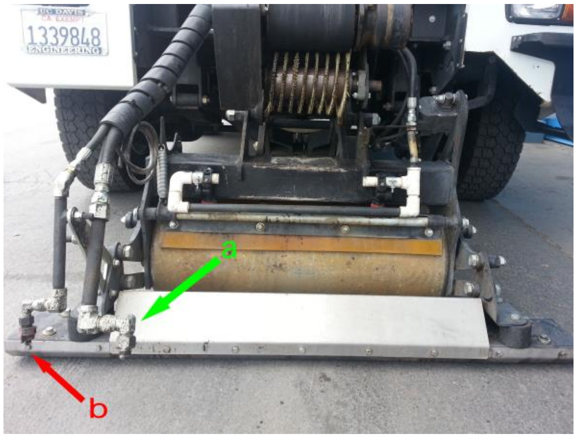


Figure 33. Spall Preparation Tools

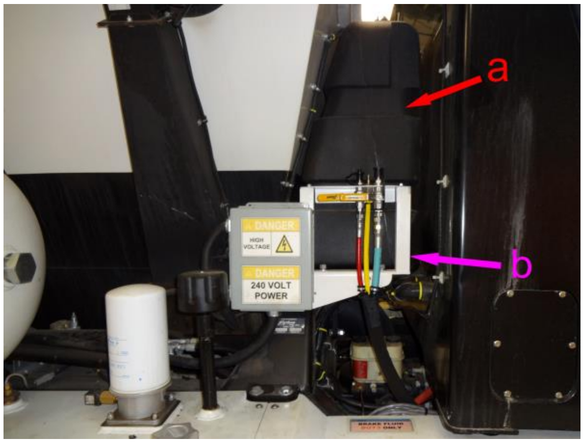


Figure 34. Urethane Disconnect

Spall Patching Preparation in Moving Closures

A prescribed procedure exists to properly prepare a spall for urethane gravel patching to achieve a permanent repair. The fundamental preparation method involves the cutting of square and perpendicular edges on the spall, as well as removing all dust and loose debris. This level of roadwork consumes a considerable amount of time, but inadequate preparation usually results in a near-term loss of adhesion to the PCC and a risk of ejecting the patch from the spall. Neither the access nor the time required to properly prepare the spall is achievable in the proposed Python PHP moving closure spall repair operation. Only a compressed air blast, currently available on the Python PHP machine, serves as a preparation tool to remove loose dust from the spall (location a in Figure 33). Therefore, the conventional urethane repair method requires modification in order to facilitate the Python PHP's moving closure spall repair operation. Caltrans maintenance has been experimenting with the use of a two-component, rapid-setting primer in areas where concrete spall sawing and loose debris removal is impractical due to time constraints. The primer creates a strong bond with the concrete, and it succeeded in retaining urethane gravel patches in mainline highway spalls with minimal PCC preparation during highway trials. Caltrans standards endorse a wide variety of materials for permanent, partial-depth PCC repairs. For example, Caltrans maintenance crews in District 11 often use POLYQuik® FastPatch for urethane gravel patching and POLYQuik® POLYPRIME as the matched pavement primer when sawing and chipping is not feasible. Both materials are distributed by the Willamette Valley Company and are Caltrans approved for partial and full depth spall repairs. Since Caltrans stopped using the tack oil system on the Python PHP, it could be replaced with the urethane primer equipment, and the tack oil spray nozzle could be replaced with a mixing tube (location b in Figure 33). If Caltrans prefers to retain the tack oil system (location a in Figure 34), the urethane primer equipment could be added to the urethane hopper insert.

Python PHP Modification for Urethane Patching

The functional capability of the Python PHP machine could potentially be extended to include the application of fast-setting urethane patch materials, producing a durable PCC spall repair. The AHMCT Research Center was working with Willamette Valley Company to adapt and integrate their POLYQuick® FastPatch Meter Dispensed/Gravel Extended Kit (MD/GC) equipment into the Python PHP machine until the research was discontinued. The FastPatch urethane system separates into three elements on the Python PHP machine: a urethane module inserted into the Python PHP asphalt hopper, a mixing and application device mounted to the patching head, and an umbilical that connects the two through the arm. One of the more challenging aspects of adding the functionality relates to routing the umbilical material hoses, the compressed air hose, the electrical control cable, and the primer hoses through the telescoping arm and existing cable carriers. These hoses remain permanently installed on the Python PHP machine, so incorporating an accessible disconnect point on the side of the Python PHP equipment would facilitate the conversion between asphalt and urethane (location b in Figure 34).

The urethane module contains the resin and catalyst tanks, the air compressor, the material pumps, and the pea gravel filler (Figure 35). The design of the urethane module permits it to slip down flush into the Python PHP asphalt hopper. This design allows the hopper doors to be closed, leaving a six-inch gap to maintain an exit passage for the umbilical. Due to the angular positioning of the Python PHP hopper, the urethane module must share the same shape, which leaves the module unstable when standing alone out of the hopper. Therefore, a module stand is required to stabilize the module when it is out of the hopper for maintenance and storage (Figure 36). Ideally, Caltrans would use standard forklifts or loaders normally found in Caltrans maintenance yards to load the urethane module into the Python PHP hopper. The greatest vertical height measured in the San Jose maintenance station occurred with their loader with bucket fork attachments, which clear approximately thirteen feet (Figure 37). Since the top rim of the Python PHP hopper measures approximately twelve feet high, the loader must be able to attach to the bottom of the urethane module to attain the necessary height for insertion. However, in order to lower the module down into the hopper, the urethane module must be hung from the top. Therefore, the design of the module stand also must allow the stand to act as a mechanism to facilitate the insertion of the urethane module into the hopper using an ordinary Caltrans bucket loader.

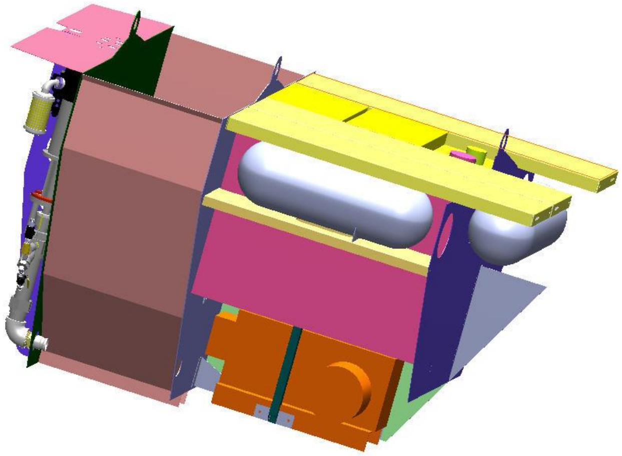


Figure 35. Python PHP Urethane Module

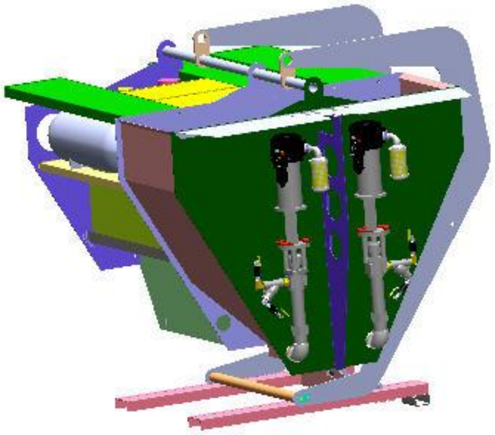


Figure 36. Urethane Module Stand

The base of the urethane module stand contains rectangular tubing, which functions as fork extensions that attach securely to bucket loader fork attachments. From the stand’s base, support arms rotate from an over-center storage position through an out and down lowering motion to the down position in the hopper. A hydraulic lift cylinder drives the motion powered by a 13 Direct

Current Voltage (Vdc) hydraulic pump unit mounted on the stand's base and controlled by a wireless remote controller. Inserting the urethane module inside the Python PHP hopper involves lifting the module unit to a preset position above and adjacent to the Python PHP hopper. Then, the operator extends the hydraulic lift cylinder that lowers the module down into the hopper (Figure 38). The lifting pipe slides out and disconnects the hopper stand. Connecting the umbilical and plugging the air compressor's electrical power into the generator fully links the urethane module to the Python PHP machine. The module contains a specific compartment for the management of the pea gravel, which stays open to the material feed auger screw on the bottom (location "a" in Figure 39). The standard Python PHP asphalt feed system feeds the pea gravel to the urethane patch. Since the partition stays open on the bottom, the gravel must be emptied via the material belt before removing the urethane module from the hopper. The pea gravel used in urethane gravel is specially graded and washed, so the aggregate should be collected and reused in a custom designed bucket insert, which also assists with the loading of the aggregate when using conventional bucket loaders.

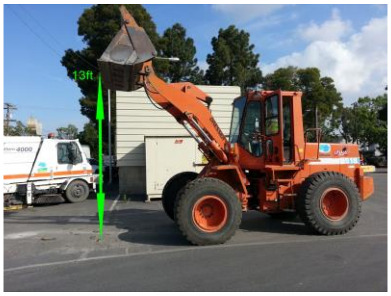


Figure 37. Bucket Loader

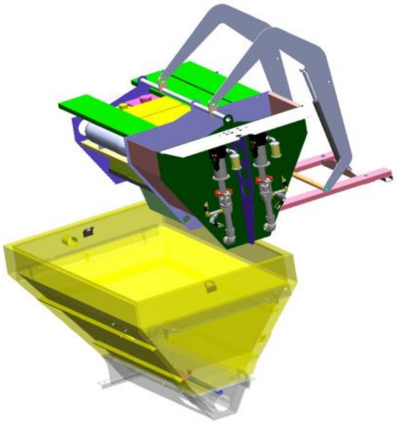


Figure 38. Inserting Module

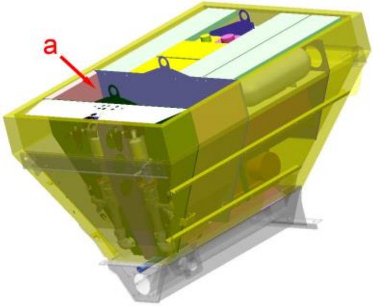


Figure 39. Module Inserted

The end of the Python PHP telescoping arm contains sufficient mounting points to easily attach the urethane mixing and application components. These components clamp onto the patching head and remain attached and out of the way in the asphalt patching configuration of the Python PHP machine. The umbilical urethane hoses routed from the hopper module connect to the metering block, which clamps to the compaction roller linkage under the patching head (Figure 40). The metering block pumps the urethane to the dispensing head in the correct proportions. Adjusting the air pressure to the metering head controls the urethane flow rate. The dispensing head, a simple on/off valve, controls the urethane’s flow through the attached mixing nozzle. The mixing head and nozzle mount to the end of the injection linkage, which provides both positional extension and a vertical translation for the nozzle. The preliminary design of the injection linkage simply clamps over the case of the Python PHP material belt drive hydraulic motor (location a in Figure 41). Note that Figure 41 displays this partial model, with the second scissor mechanism hidden to improve visibility and clarity. The scissor mechanism design extends the urethane applicator assembly out and down to a spot two feet out from the center of the material belt and four inches above the pavement. This area approximates where the pea gravel rolls off the material belt and lands on the pavement. Applying the urethane binder and aggregate matrix at this common spot promotes a natural blending of these two components in the spall. In addition, the nozzle can be moved vertically six inches, enabling the urethane to inject directly into the patch matrix as needed (location b in Figure 41). In the retracted position, the applicator assembly closes into a compact position out of the way of the Python PHP material belt (Figure 42). Mounted below the nozzle in the retracted position, a disposable spoils bucket catches the urethane expelled during the nozzle refreshment stage.

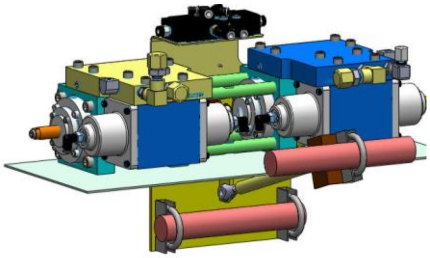


Figure 40. Metering Block

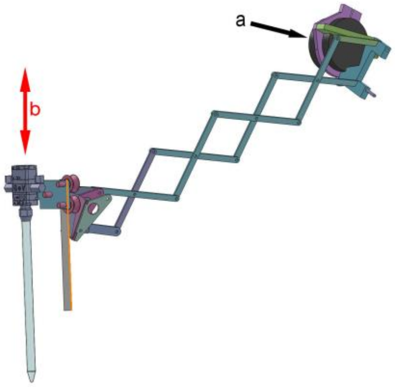


Figure 41. Injection Linkage

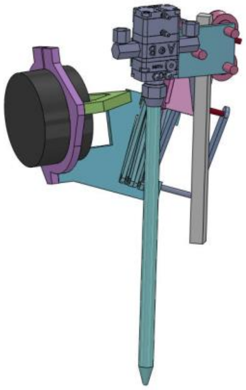
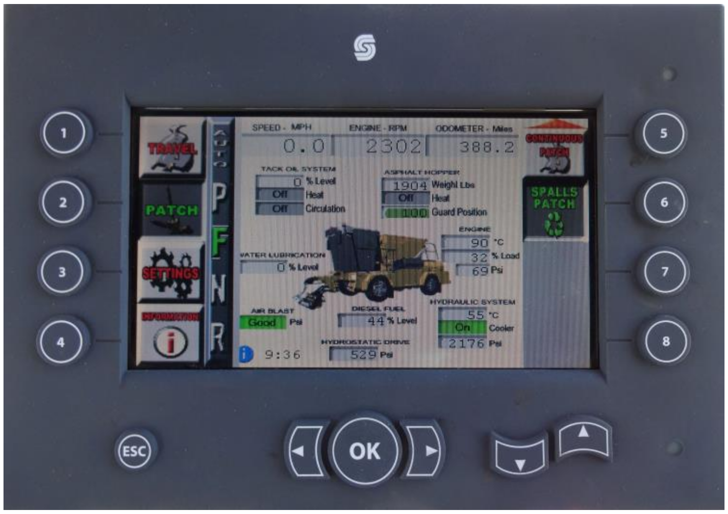


Figure 42. Retracted Linkage

<u>Spall Patching Operation in Moving Closures</u>

Using the Python PHP as a urethane based spall patching machine may retain all the benefits of mainline moving closure patching operations. The traffic control measures developed in Caltrans District 4 may apply to urethane patching operations, if equivalent patching times can be attained. Quick preparation procedures, including primer coating and fast-setting urethane mixtures, will all be fundamental in reducing urethane patching times sufficiently to support safe moving closure operations on the mainline. In theory, once the Python PHP machine is configured and loaded for urethane patching operations, it can be parked on stand-by, ready to be deployed at short notice to patch spalls. To patch, the dedicated operator drives the Python PHP machine to a safe location near the beginning of the work site and installs a fresh mixing nozzle. In the cab, the operator selects the **SPALL PATCH** mode on the control panel (Figure 43). The control panel displays an informational screen describing how the Python PHP operational control changes when dispensing urethane (Figure 44). In this example, the **#3-EPOXY ON/OFF BUTTON** on the joystick controller activates the pumping of urethane through the nozzle. After the establishment of urethane flow through the mixing nozzle, the Python PHP controller automatically purges the mixing nozzle at regular intervals with new urethane to prevent clogging. Now, the Python PHP machine is ready to be driven out on the highway to begin patching operations. The patching crew and traffic control personnel establish the moving closure, and the operator drives the Python PHP machine into the closed lanes and patrols for spalls. The Python PHP machine and the traffic control vehicles can accelerate up to and travel at near highway speeds on the mainline between repairs.



SPEED - MPH 0.0

ENGINE - RPM 2302

ODOMETER - Miles 388.2

TACK OIL SYSTEM
0 % Level
Off Heat
Off Circulation

ASPHALT HOPPER
1904 Weight Lbs
Off Heat
100 Guard Position

ENGINE
90 °C
32 % Load
69 Psi

WATER LUBRICATION
0 % Level

AIR BLAST
Good Psi

DIESEL FUEL
44 % Level

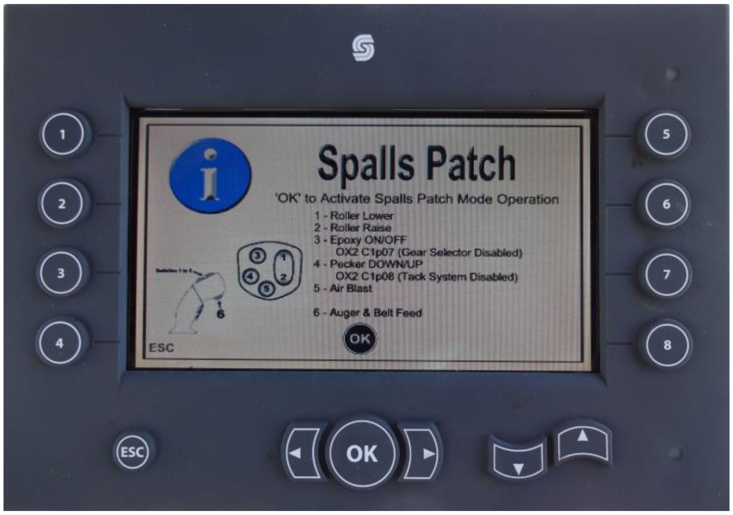
HYDRAULIC SYSTEM
55 °C
On Cooler
2176 Psi

HYDROSTATIC DRIVE
529 Psi

9:36

Figure 43. Spall Repair Screen

The Python PHP machine leads the traffic control procession and stops in front of spalls in need of repair. With the standard equipment compressed air nozzle mounted on the Python PHP patching head, the operator clears loose dust and debris clear from the spall. Since moving closure patching operations involve a less than ideal degree of spall preparation, a coating of primer must be applied to the spall to promote urethane adhesion. The two-part primer is applied through a secondary mixing nozzle attached to the Python PHP patching head on the end of the telescoping boom. Using the joystick, the operator moves the applicator around the spall and applies an even coating of primer. After a set time, the dedicated operator similarly applies the urethane as a coating in the spall before adding the pea gravel matrix. For maximum patch strength and adhesion, mixing the gravel and the urethane uniformly is important. The method to accomplish this is to apply urethane at the bottom of the spall, urethane on top of the gravel filler, and then inject urethane directly into the gravel filler by randomly poking the nozzle into the gravel patch. To apply pea gravel, the operator presses the **#6-AUGER & BELT FEED BUTTON** on the joystick, which runs gravel off the material belt and into the spall. The operator works the urethane nozzle and gravel filling in concert to fill the spall, striving to ensure the pea gravel remains immersed in urethane. The urethane nozzle may also be driven down into the gravel with the **#4-PECKER BUTTON** on the joystick to inject urethane directly into pockets of dry pea gravel. A plastic cover on the rake provides the operator with a means of tooling the surface of the patch if necessary. The Python PHP machine and all traffic control vehicles avoid driving over the fresh patch to provide an undisturbed set time before being exposed to the trailing highway traffic.



Spalls Patch

'OK' to Activate Spalls Patch Mode Operation

1 - Roller Lower
2 - Roller Raise
3 - Epoxy ON/OFF
OX2 C1p07 (Gear Selector Disabled)
4 - Pecker DOWN/UP
OX2 C1p08 (Tack System Disabled)
5 - Air Blast

6 - Auger & Belt Feed

Figure 44. Spall Controls Screen

Conclusion/Python PHP Machine Current Status

Caltrans District 4 maintenance crews, in partnership with the AHMCT Research Center, created an innovative and highly efficient operation that utilizes the Python PHP machine to patch mainline highway potholes in moving lane closure operations. Automation, the key element, enables a worker to patch potholes from inside the vehicle cab, which accounts for the high level of worker protection and efficiency achieved. The equipment acquired for the semi-automated pothole patching research was a prototype version of the Python PHP machine developed specifically for Caltrans' use. Following a successful year-and-a-half deployment in the District 4 San Jose maintenance yard performing preventative maintenance and urgent pothole repair operations, the Python PHP machine was removed from service in July 2013 due to safety and handling concerns expressed by Caltrans Headquarters Maintenance. The AHMCT Research Center received the following list of Python PHP machine deficiencies explaining the machine's removal from operation:

1) Unsatisfactory steering: Slack in the Python PHP steering box reduced the steering response in highway driving and required the driver to over-steer vehicle turning.
2) Stiff suspension: The Python PHP machine's short wheelbase to load capacity ratio produced a rough ride when empty.
3) Insufficient Gross Vehicle Weight (GVW) capacity: The Python PHP machine can become slightly overloaded when the hopper is fully loaded with hot asphalt.
4) Intermittent cab air conditioning: Operator cab air conditioner operated intermittently.
5) Non-certified operator cab: The Python PHP operator cab was not federally DOT-certified and, specifically, lacked the required high-back bucket seat and three-point safety restraint.
6) Incomplete chassis certification: The Python PHP vehicle frame and identification plate do not meet the required Federal Motor Vehicle Safety Standards–49 Code of Federal Regulations (CFR) Part 567. The Python PHP machine also lacked the 17-digit Vehicle Identification Number (VIN) that meets National Highway Traffic Safety Administration regulation 49 CFR Part 565 requirements.

The expressed handling and safety concerns from Caltrans Headquarters Maintenance led to the Python PHP machine's removal from service and return to UC-Davis. District 4

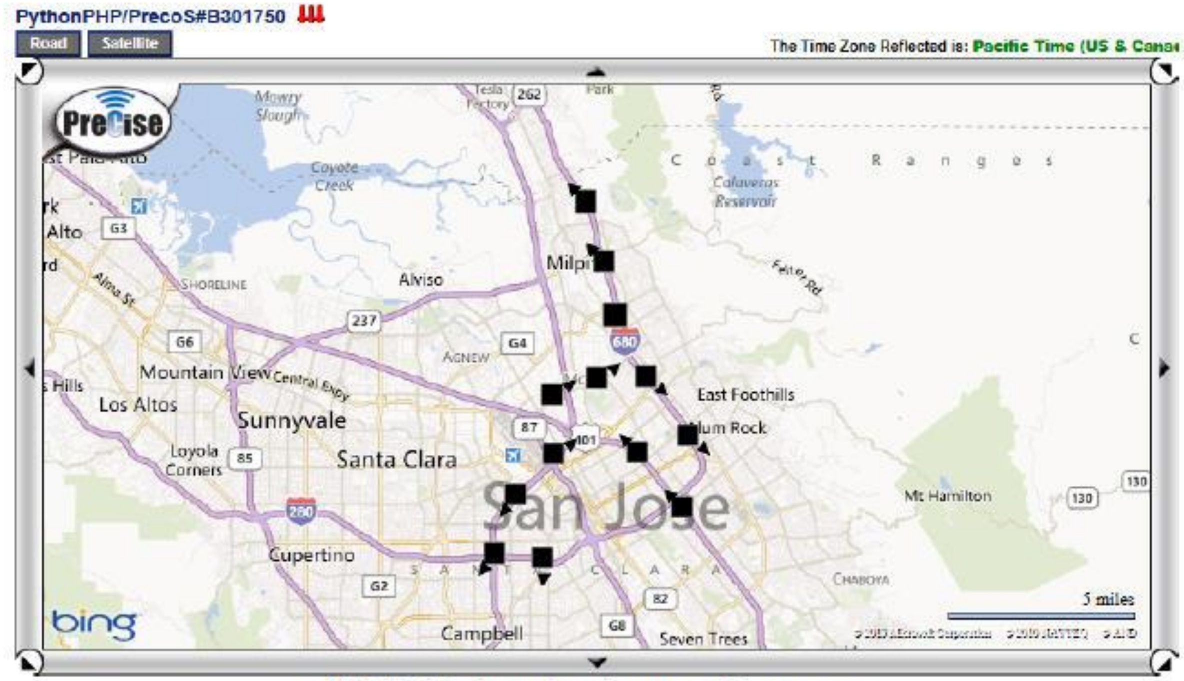
Maintenance fully integrated the PHP into their work schedule prior to its removal from service. The Python PHP machine remains functional and located at UC-Davis, and Caltrans District 4 workers returned to patching potholes manually on foot. The AHMCT Research Center continued to develop the Caltrans urethane PCC spall patching upgrade task at UC-Davis. However, in February 2014, Caltrans requested that the AHMCT Research Center cease work on the active research task that would develop the urethane spall repair capabilities for the Python PHP machine.

Future Development

Caltrans District 4 Maintenance remains interested in obtaining a Python PHP machine for moving closure mainline pothole repair operations. Python Mfg. is currently developing a new generation Python PHP machine with many new features, including mitigation measures for all the deficiencies found in the current Python PHP machine, as described by Caltrans Headquarters Maintenance. In addition, Python Mfg. plans to incorporate all of the upgraded features developed by the AHMCT Research Center to support moving closure operations.

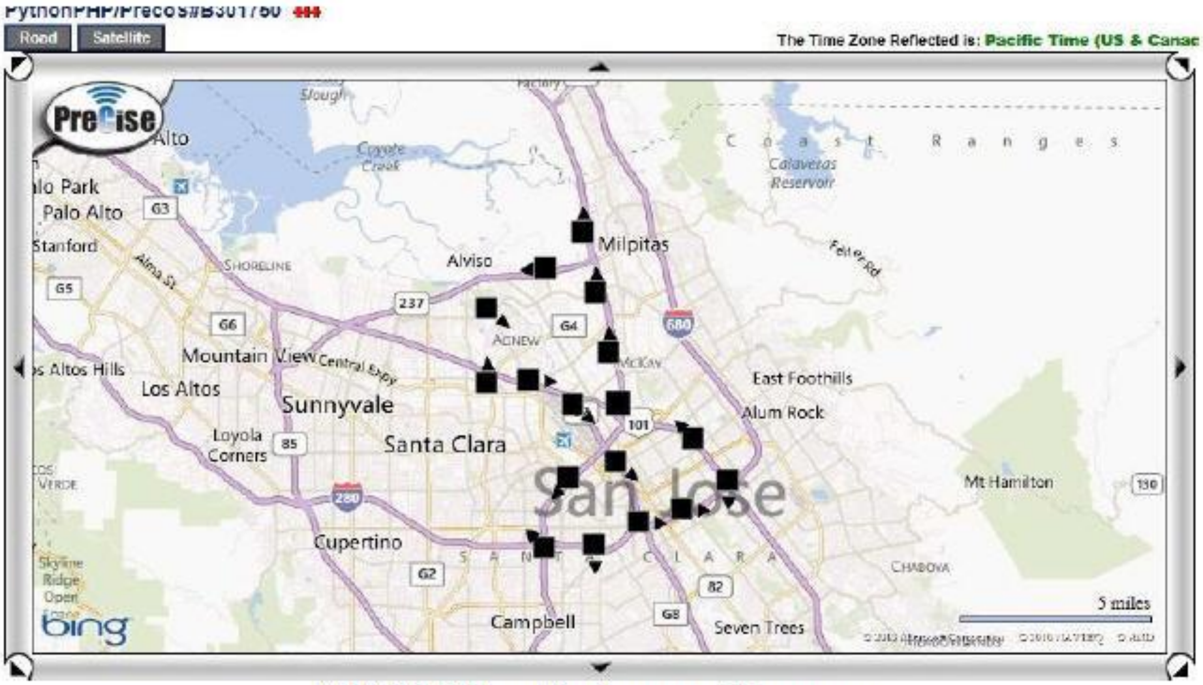
Appendix: Python PHP Machine Patching Logs

Caltrans District 4 Python PHP Pothole Patching Reports March 3, 2013 – June 11, 2013



AHMCT Python Deployment Summary

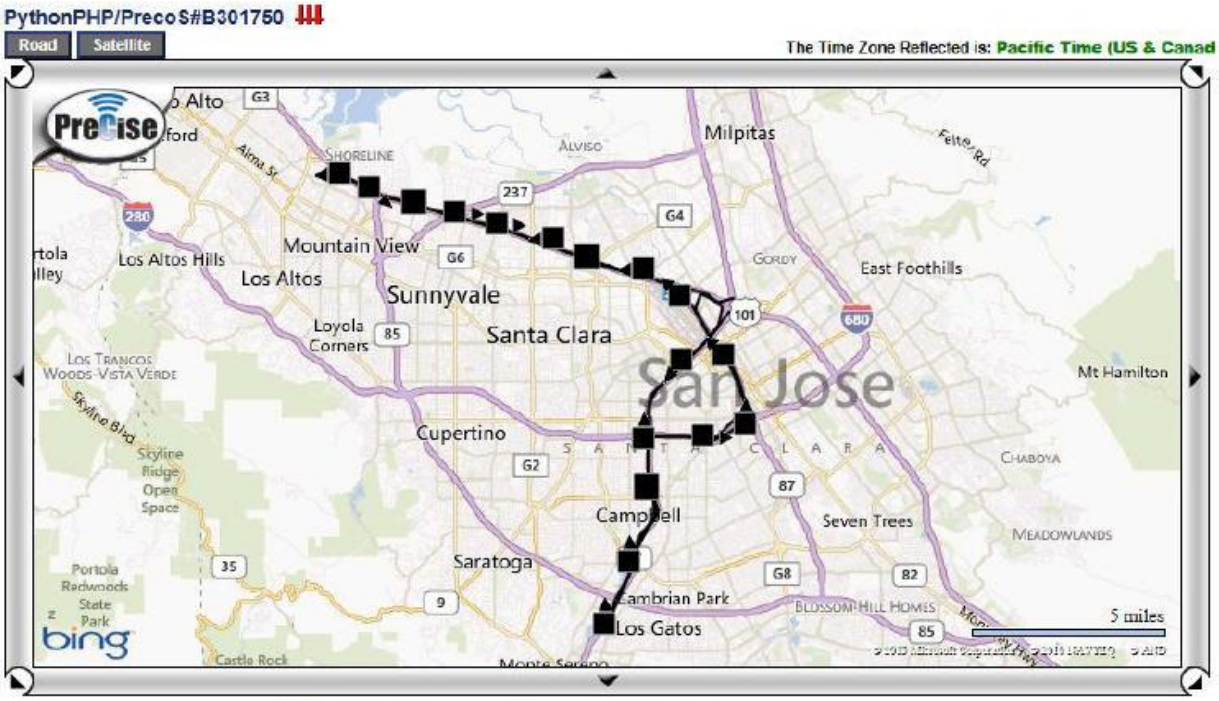
Date:	3-12-2013
Start time:	8:30 AM
District:	4 San Jose, Ca.
Roadway:	Hwy 680 north and south
Type of patching:	Pot holes and longitudinal voids on PCC and AC
Supervisor:	Paul Salaiz
Operator:	Jorge Martinez
Batch Plant:	Reed & Graham San Jose, Ca.
Total hot mix loaded:	4.5 tons (9000 lbs.)
Total hot mix used:	100%
Total time patching:	1 hour 45 minutes
Type of road closure:	Rolling
CHP present:	2 cruisers
Hot mix waste dumped:	0 lbs.
Tack oil:	Not used as recommended by Python
Reported vehicle issues:	Belt tracking off slightly, rear roller. Reverse gear sluggish.
Notes:	This was a training deployment for District 4's new primary operator. Jorge did an exceptional job for having very limited experience.



AHMCT Python Deployment Summary

Date:	3-13-2013
Start time:	10:30 AM
District:	4 San Jose, Ca.
Roadway:	Hwy 101 north and south
Type of patching:	Pot holes and longitudinal voids on PCC and AC shoulder/transitions
Supervisor:	Paul Salaiz
Operator:	Jorge Martinez
Batch Plant:	Granite Construction Santa Clara, Ca.
Total hot mix loaded:	4.9 Tons (Approximately 10,000 lbs.)
Total hot mix used:	100%
Total time patching:	2 hours
Type of road closure:	Rolling 2 shadow vehicles
CHP present:	2 cruisers
Hot mix waste dumped:	0 lbs.
Tack oil:	Not used as recommended by Python
Reported vehicle issues:	None

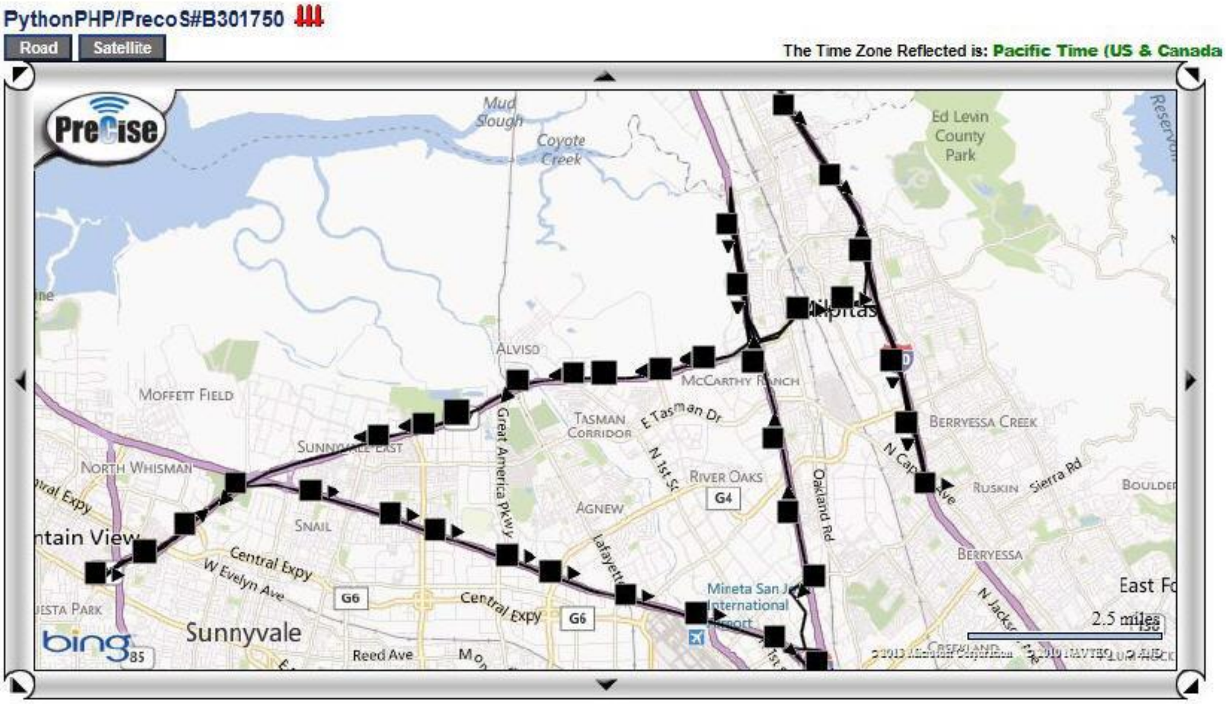
Notes: Caltrans attempted to reload the PHP to continue patching but couldn't get any more hot mix that afternoon. All subsequent hot mix produced at this plant was ear marked for a private contractor's special project. Inspection after cleanup revealed conveyor belt tracking is unacceptable and will eventually destroy critical moving parts, energy chain and or the conveyor belt. The PHP is down until further notice as a precaution. Python has responded to reported issues with a replacement parts kit for correction. Parts should arrive soon at UCD and will be installed thereafter.


PythonPHP/PrecoS#B301750

AHMCT Python Deployment Summary

Date:	5-07-2013
Start time:	10:20 AM
District:	4 San Jose, Ca.
Roadway:	Hwy 101 south, 87, 17
Type of patching:	Pot holes and longitudinal voids on PCC and AC shoulder
Supervisor:	Paul Salaiz
Operator:	Jorge Martinez
Batch Plant:	Granite Construction Santa Clara, Ca.
Total hot mix loaded:	4.9 Tons (Approximately 10,000 lbs.)
Total hot mix used:	100%
Total time patching:	3 hours
Type of road closure:	Rolling 2 shadow vehicles
CHP present:	2 cruisers
Hot mix waste dumped:	0 lbs.
Tack oil:	Not used as recommended by Python
Reported vehicle issues:	Break lights in/op

Notes: Break lights were in/op. Investigated probable cause and performed basic trouble shooting: Checked fuses, break switch and wiring. Initial investigation was inconclusive due to a lack of proper tools and time constraints. More trouble shooting of this problem will be required. Supervision decided to continue with deployment ops since the PHP was loaded with 4.9 tons of fresh hot mix. Shadow vehicle drivers were advised to be extra observant. Arvern Lofton of DRI and Dale Greep of DOE, META was present for deployment ops and rode along with the operator during this deployment. New rear roller mod appears to be functioning as designed and without issue.

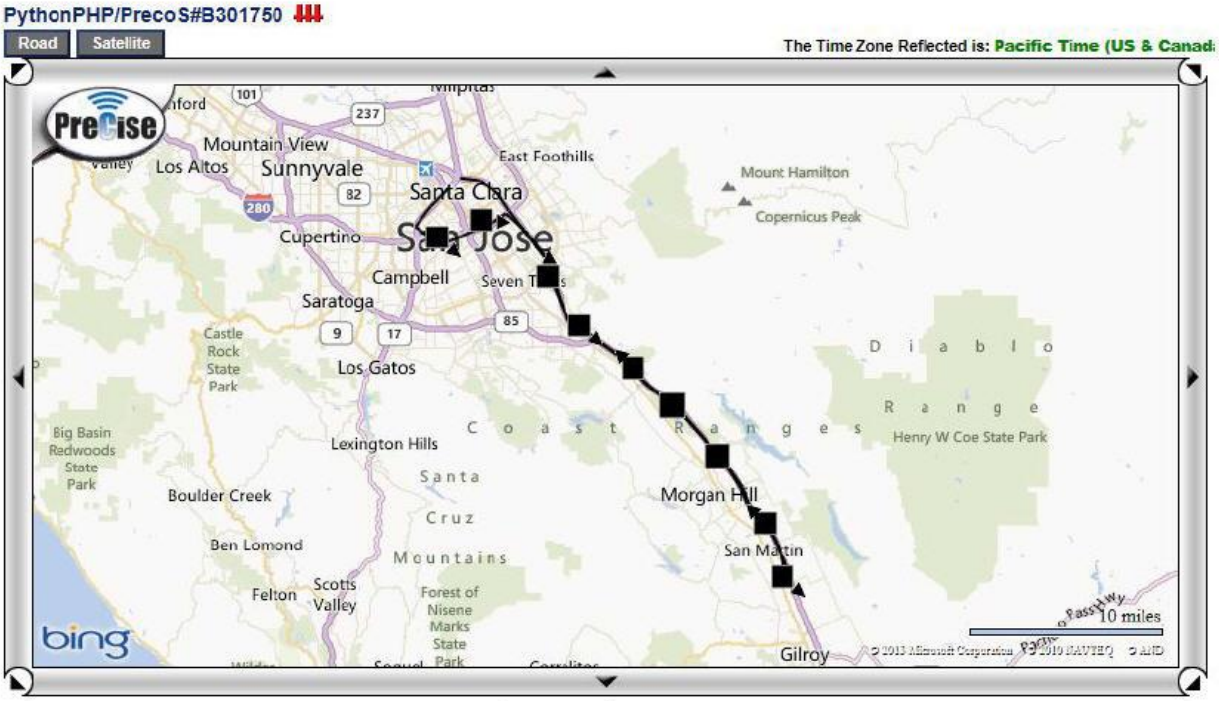


The Time Zone Reflected is: Pacific Time (US & Canada

AHMCT Python Deployment Summary

Date:	5-22-2013
Start time:	8:30 AM
District:	4 San Jose, Ca.
Roadway:	Hwy 87, 237, 680, 880
Type of patching:	Pot holes and longitudinal voids on PCC and AC
Supervisor:	Paul Salaiz
Operator:	Jorge Martinez
Batch Plant:	Reed and Graham
Total hot mix loaded:	4.5 tons (9000 lbs.)
Total hot mix used:	100%
Total time patching:	2 hours. Filled approx. 75 pot holes & 7 longitudinal voids
Type of road closure:	Rolling, 3 Shadow Trucks, MAZEEP
CHP present:	2 cruisers
Hot mix waste dumped:	0
IMMS work order #:	3042693, 3042698
Reported vehicle issues:	None

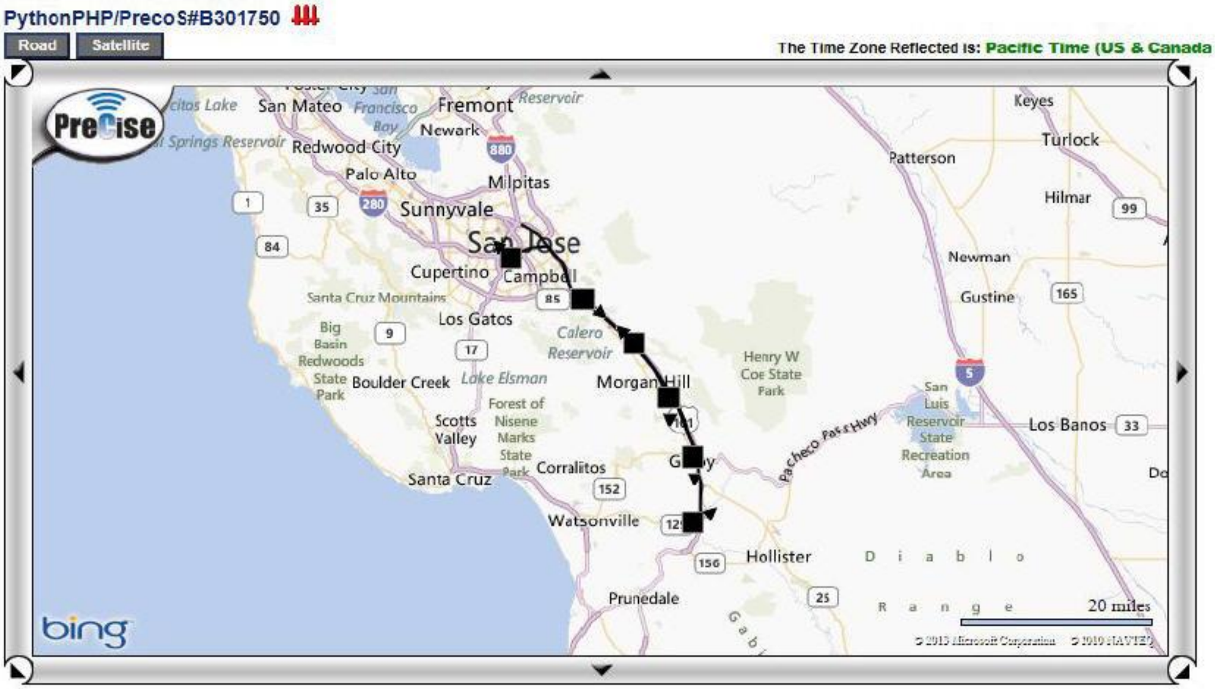
Notes: Paul Salaiz initiated the training of a backup PHP primary operator. Brandon Diaz rode along with Jorge during this deployment as an observer as part of his training. All previously installed parts are operating without issue. The replacement belt is functioning as designed and will continue to be monitored for susceptibility to D-lamonine. Belt tracking appears to be normal and there are no indications of abnormal or pre mature belt wear. The A.C. continues to work intermittently and will be serviced ASAP. Dale Greep from DOE was not present. The Break light switch alignment was corrected and is now working as designed.



AHMCT Python Deployment Summary

Date:	5-29-2013
Start time:	10:00 AM
District:	4 Gilroy, Ca.
Roadway:	Hwy 101 South
Type of patching:	Pot holes and longitudinal voids on PCC and AC
Supervisor:	Paul Salaiz
Operator:	Jorge Martinez & Brandon Diaz (Operator trainee)
Batch Plant:	Reed and Graham
Total hot mix loaded:	5 tons (10,000 lbs.)
Total hot mix used:	100%
Total time patching:	2 hours. Filled approx. 66 pot holes & 31 longitudinal voids
Type of road closure:	Rolling, 2 shadow Trucks, MAZEEP
CHP present:	1 cruiser
Hot mix waste dumped:	0
IMMS work order #:	3045274, “Python”
Reported vehicle issues:	Boom roller water system obstruction

Notes: Conducted trouble shooting of the water system over the phone with the crew in San Jose. The obstruction was caused by calcification due to use of hard water. Spray nozzles were removed and cleaned. Other involved plumbing was also cleaned and blown out with air. Operators will continue to monitor the water system.



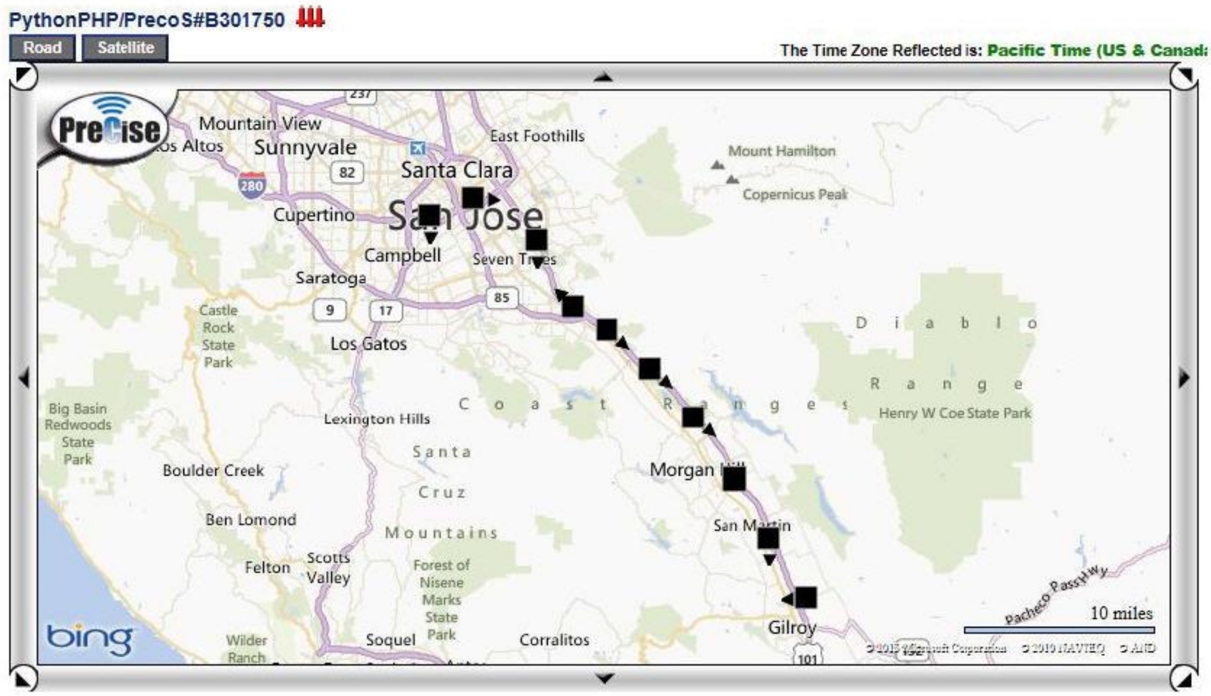
PythonPHP/PrecoS#B301750

The Time Zone Reflected is: Pacific Time (US & Canada

AHMCT Python Deployment Summary

Date:	5-30-2013
Start time:	10:00 AM
District:	4 Gilroy, Ca.
Roadway:	Hwy 101
Type of patching:	Pot holes and longitudinal voids on PCC and AC
Supervisor:	Paul Salaiz
Operator:	Brandon Diaz (Operator trainee)
Batch Plant:	Reed and Graham
Total hot mix loaded:	5 tons (10,000 lbs.)
Total hot mix used:	100%
Total time patching:	3 hours. 134 total patches, combination of pot holes and longitudinal voids
Type of road closure:	Rolling, 2 Shadow trucks, MAZEEP
CHP present:	1 cruiser
Hot mix waste dumped:	0
IMMS work order #:	3045274 "Python"
Reported vehicle issues:	None

Notes: The rear passenger side amber strobe light works intermittently. Operators will continue to monitor the watering system.



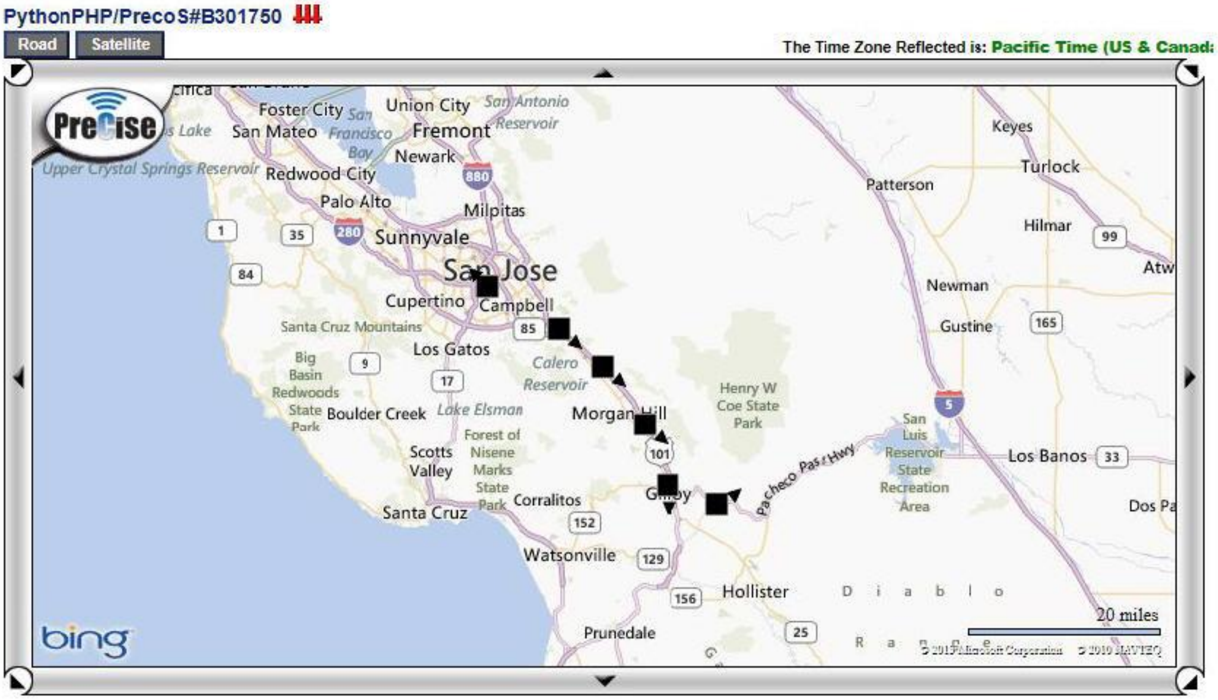
PythonPHP/PrecoS#B301750

The Time Zone Reflected is: Pacific Time (US & Canad

AHMCT Python Deployment Summary

Date:	6-04-2013
Start time:	10:00
District:	4 Gilroy, CA
Roadway:	Hwy 101 North
Type of patching:	Pot Holes, longitudinal voids
Supervisor:	Paul Salaiz
Operator:	Brandon Diaz
Batch Plant:	Reed and Graham
Total hot mix loaded:	5 tons (10,000 lbs.)
Total hot mix used:	100%
Total time patching:	1 hour 30 minutes. 140 total patches, pot holes and shoulder voids
Type of road closure:	Rolling, 2 shadow trucks, MAZEEP
CHP present:	1 cruiser
Hot mix waste dumped:	0
IMMS work order #:	Python
Reported vehicle issues:	None

Notes:



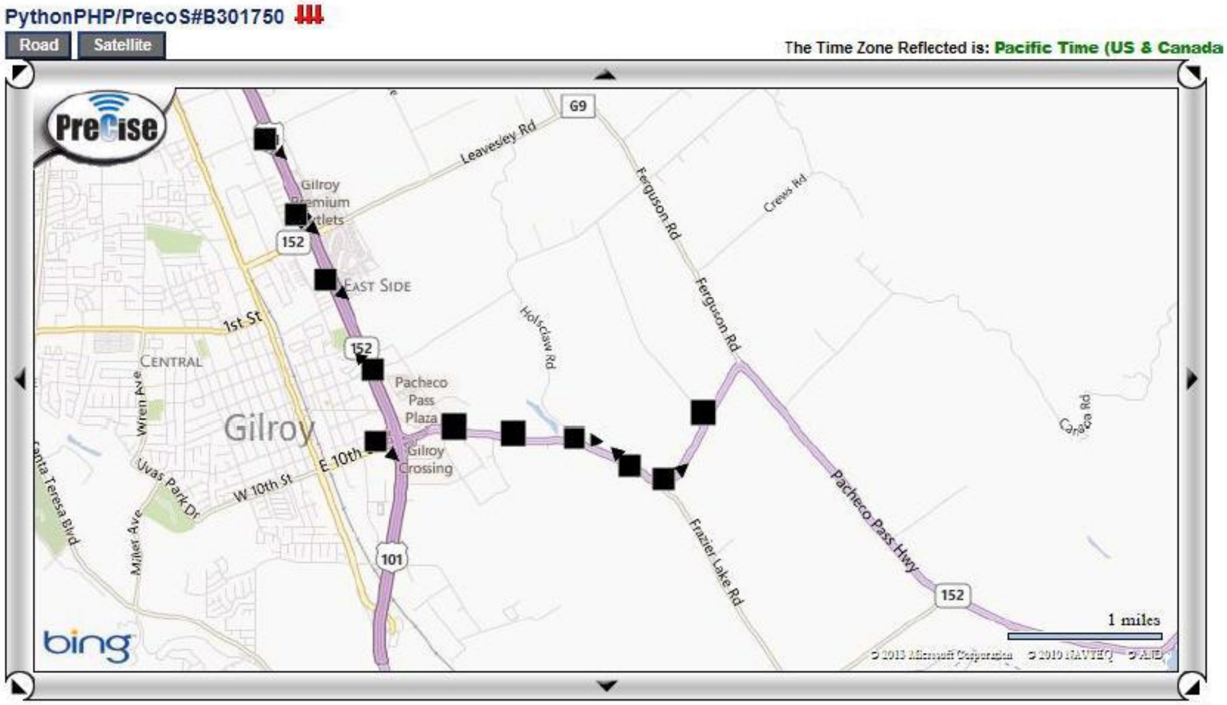
PythonPHP/PrecoS#B301750

The Time Zone Reflected is: Pacific Time (US & Canad

AHMCT Python Deployment Summary

Date:	6-06-2013
Start time:	10:30
District:	4 Gilroy, CA
Roadway:	Hwy 152 East
Type of patching:	Pot holes and shoulder voids
Supervisor:	Paul Salaiz
Operator:	Jorge Martinez
Batch Plant:	Reed and Graham
Total hot mix loaded:	5 tons (10,000lbs.)
Total hot mix used:	5%
Total time patching:	20 Minutes
Type of road closure:	Rolling 1 shadow truck, MAZEEP
CHP present:	1 Cruiser
Hot mix waste dumped:	95%
IMMS work order #:	Python
Reported vehicle issues:	None

Notes: The hot asphalt mix loaded at the batch plant was determined to be bad. The hot mix contained a disproportionate amount of oil and sand and not enough aggregate. Paul Salaiz determined the mix was suspect and aborted deployment ops after the PHP operator made a few patches on Hwy 152 just east of Hwy 101. The patches would not set up and the hot mix material was sticking to the compaction roller. The remaining hot mix was disposed of appropriately.


PythonPHP/PrecoS#B301750

The Time Zone Reflected is: Pacific Time (US & Canada

AHMCT Python Deployment Summary

Date:	6-11-2013
Start time:	10:00
District:	4 Gilroy, CA
Roadway:	Hwy 152 East and West
Type of patching:	Pot holes, shoulder and longitudinal voids
Supervisor:	Paul Salaiz
Operator:	Jorge Martinez
Batch Plant:	Reed and Graham
Total hot mix loaded:	4.5 Tons (9,000 Lbs.)
Total hot mix used:	100%
Total time patching:	1 hour 45 minutes. 87 total patches, pot holes and longitudinal voids
Type of road closure:	Rolling, 1 shadow truck, MAZEEP
CHP present:	1 Cruiser
Hot mix waste dumped:	0
IMMS work order #:	Python
Reported vehicle issues:	None

Notes: Most of Highway 152 is a two lane highway. The Caltrans patching crew would patch for 5 minutes and then pull of the road to let traffic clear to minimize traffic congestion. This method was effective and used in both east and west bound directions and had minimal effect on production.

The operator reported that the brake pedal was sticking when the vehicle was being cleaned in the yard after patching ops. Upon further inspection, OEM John Deere cab mounts located on the front end of the vehicle had collapsed. AHMCT staff parked the Python until cab mount were changed out. On board system programming was updated. The Python was taken to a local vendor for a BIT inspection and preventative maintenance since no patching ops were scheduled for 6/17-21/2013.

References

[1] “The Mile Marker: A Caltrans Performance Report,” Caltrans, Vol. 1, Issue 1, January 2014.

[2] Pothole Repair Tech Brief, FHWA – U.S. Department of Transportation
FHWA Contact: Monte Symons, 202-493-3144
ERES Consultants, Inc. Contract No. DTFH-93-C-00051, November 1999
https://www.fhwa.dot.gov/publications/research/infrastructure/pavements/ltpp/99202/99202.pdf

[3] Evaluation of Pothole Patching Equipment and Processes
AHMCT Research Center – California Department of Transportation
Report CA12-1738, D. Bennett, V. Reveles, S. Velinsky, June 2012
http://www.dot.ca.gov/newtech/researchreports/reports/2012/final_report_task_1738.pdf

[4] “California Transportation by the Numbers: Meeting the State’s Need for Safe and Efficient Mobility,” TRIP, A National Transportation Research Group, September 2014.

[5] Bhattacharya, B., Zola, M. P., Rao, S., Smith, K., & Hannenian, C., “Performance of Edge Drains in Concrete Pavements in California,” California State University, Chico, https://www.csuchico.edu/cp2c/documents/Library/2g-bhattacharya.pdf.

[6] “Isolated Partial Depth Concrete Repair,” *MTAG Volume II: Rigid Pavement Preservation*, 2nd ed., Caltrans Division of Maintenance, January 18, 2008.