CD.FOUNDATION

Streamlining Global Continuous Delivery

The Impact of Developer Relations Automation at Autodesk

CHALLENGE

Autodesk empowers developers and businesses to create innovative, customized applications and services through Autodesk Platform Services (formerly Forge), a cloud-based developer platform. An open source strategy compelled Autodesk to implement Continuous Delivery (CD) practices since several challenges often occur during software delivery.

The challenges that were experienced by over 12,000 developers a day and over 600,000 Jenkins builds per month included:

- **OPERATIONAL INEFFICIENCIES:** Any issues in the development or release workflow require updates and communications across many tools, resulting in inefficient and delayed resolutions.
- **CUSTOMER EXPECTATIONS:** Customers experiencing incidents or issues often expect requests to be triaged and fixed ASAP.
- **COMPLEX DEVELOPMENT PROCESSES:** Software products are intricate and extensive, involving multiple teams, technologies, and interdependent components, leading to lengthy development cycles, and slowing the release of new features and updates.
- **FEEDBACK IS SCARCE:** Customer feedback is far removed and without a solution to quickly provide context for platform incidents and issues, developers can't adapt the platform to evolving needs and industry trends.
- **COMPETITIVE LANDSCAPE:** Time to market/ release is critical, requiring the ability to accelerate software development and delivery to maintain a leading edge.

These challenges led to many feature-oriented development queries and issues that either went unanswered or subject to significant delay. Support and engineering had to wait and check for updates and monitor all communication channels for platform issues 24/7 worldwide. This setup meant the load was unmanageable during outages or maintenance. The amount of time that was spent on triaging was costly and time-consuming.

AUTODESK

CHALLENGES
Time to deliver, Platform reliability, Developer enablement

INDUSTRY
Manufacturing

LOCATION
Global

PUBLISHED
November 2023

AUTHORS
Stella Kislyuk, Senior Manager, Global Platform Engineering Developer Ecosystem

Tiffany Jachja, Engineering Manager, Global Platform Support

Yuanmin Wei, Principal Engineer

PROJECTS
Jenkins Spinnaker

Everything was manual, there were no processes in place so more time was being spent on triaging than developing the solutions to fix the root cause. These challenges frustrated developers already integrating new technologies, processes, and ways of working. All of this resulted in production release delays and—most importantly—a lack of trust between the engineering and support teams.

BY THE NUMBERS

- 12,000 developers
- 600,000 Jenkins builds per month

SOLUTION

Several macro and micro industry trends influenced Autodesk's decision to introduce Developer Relations within Autodesk. Developer Relations asks how we can empower developers by reducing the overhead in development workflows, the risk in delivering, and the operation overhead in supporting platform solutions.

1. **AUTOMATION FIRST:** The widespread adoption of DevOps practices in the industry demonstrated the benefits of close collaboration between development and operations teams. Continuous Delivery, a key aspect of DevOps, promised to automate and optimize software delivery processes.
2. **OPEN SOURCE PROLIFERATION:** The software community's increasing reliance on open source technologies allowed Autodesk to leverage existing tools and frameworks, enabling faster development and fostering collaboration with the open source community.
3. **EMERGING AND CLOUD-NATIVE TECHNOLOGIES:** The rise of AI and cloud native technologies emphasized the need for flexible and scalable software delivery methods. Continuous Delivery aligned well with these modern development paradigms and approaches.

RESULTS

After onboarding the Autodesk DevRel Automation Platform the following results were reported:

- Increased Developer Trust
- Improved Productivity

IMPROVED PRODUCTIVITY

We provided a Developer Relations Platform to streamline DevRel support processes, maximize product team productivity, and ensure engineering excellence. Autodesk DevRel Automation Platform is a distributed system implemented in a microservice architecture. The platform provides developers with self-help features, including:

- Slack-Jira Integration
- Self-Help service powered AI Bot
- Analytics

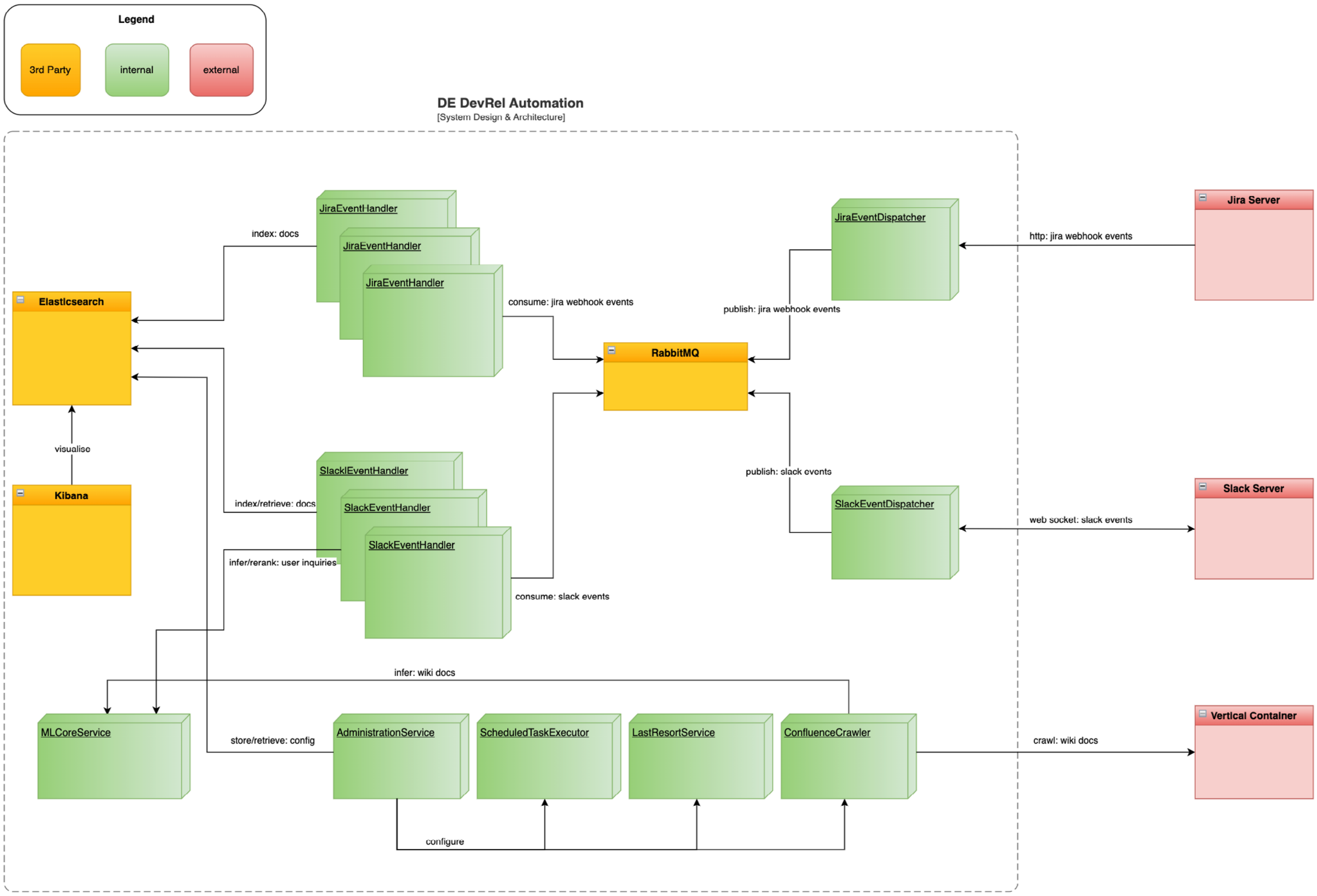
SOLUTION OVERVIEW

The platform features are customizable, allowing teams to create highly specific and effective automation for all delivery needs. Autodesk platform teams used this for their delivery needs across Spinnaker, Jenkins, and other open-source software delivery tooling.

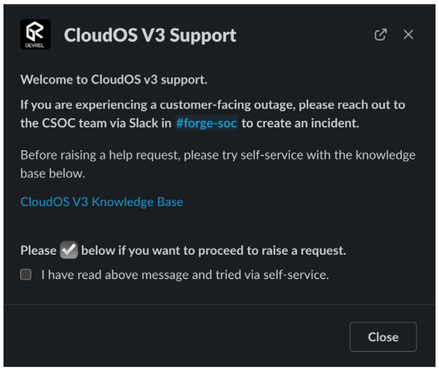
FEATURE	DESCRIPTION
Slack UI	Highly customized Slack UI with Slack forms, interactive components
Slack-Jira Integration	Bridge Slack and Jira with creating Jira issue on Slack message, comments syncing, status update notification
Load Balancing/Automated Ticket Assignment	Distribute customer request to on-call engineers on pre-defined policies
Emoji Workflow	Interact with Jira issue by emojis
CSAT Collection	Collect feedback from customer
Analytics Dashboard	Dashboard with visualizations on data collected from support process
Reporting	Scheduled reports
Self-Help AI Service	Self-service powered by NLP

DESIGN

The solution is designed and implemented as a distributed system in microservices to ensure the scalability, flexibility, and capability of handling various integration requirements.



INTEGRATION


CloudOS V3 Support

Welcome to CloudOS v3 support.

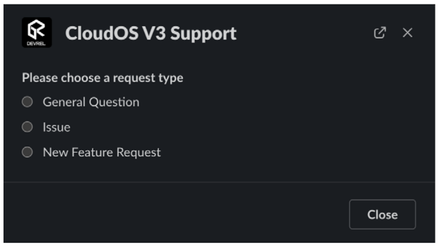
If you are experiencing a customer-facing outage, please reach out to the CSOC team via Slack in #forge-soc to create an incident.

Before raising a help request, please try self-service with the knowledge base below.

CloudOS V3 Knowledge Base

Please ✓ below if you want to proceed to raise a request.

□ I have read above message and tried via self-service.

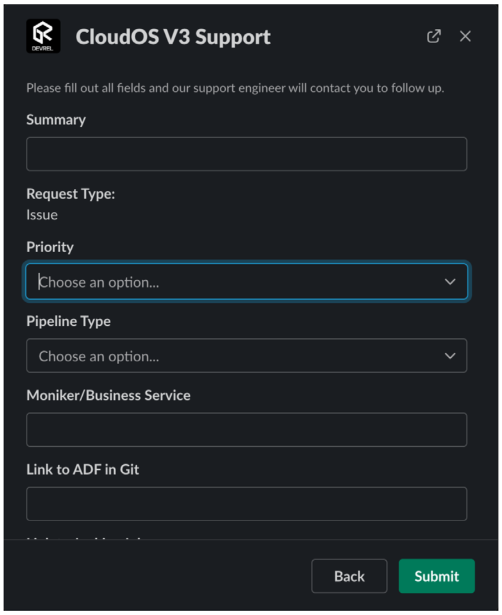

CloudOS V3 Support

Please choose a request type

○ General Question

○ Issue

○ New Feature Request


CloudOS V3 Support

Please fill out all fields and our support engineer will contact you to follow up.

Summary

Request Type:
Issue

Priority

Choose an option...

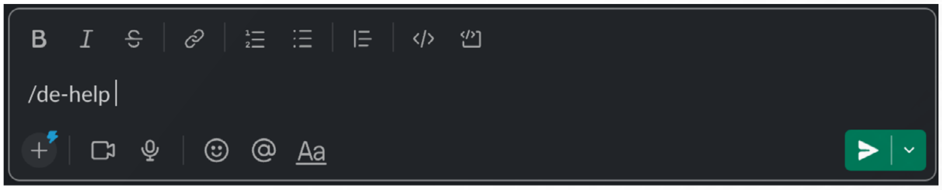
Pipeline Type

Choose an option...

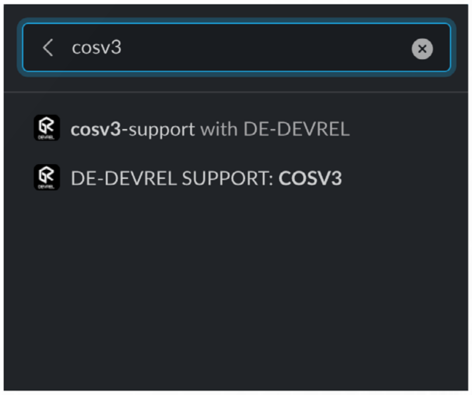
Moniker/Business Service

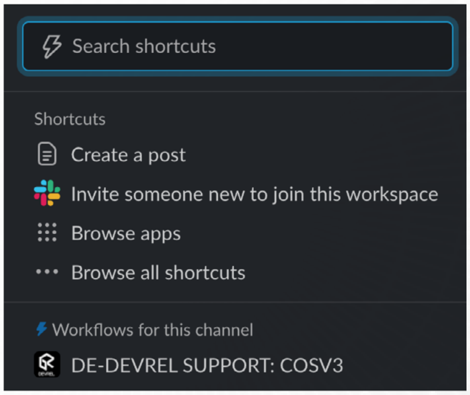
Link to ADF in Git

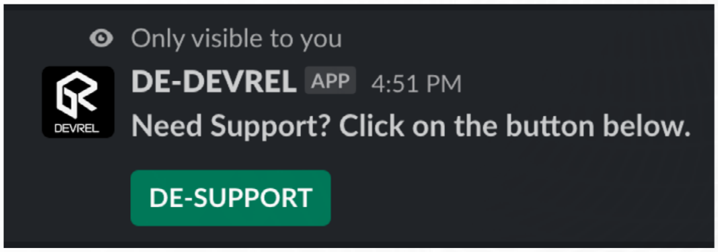
Seamless integration between Slack and Jira allows developers to issue their requests, collecting relevant incident details and input at the time of the request. Developers provide all necessary information without manual intervention from a support engineer. All updates and notifications are provided through Slack, and now, developers can provide ratings and feedback that support engineers who can then log and analyze the feedback for future improvement.



/de-help







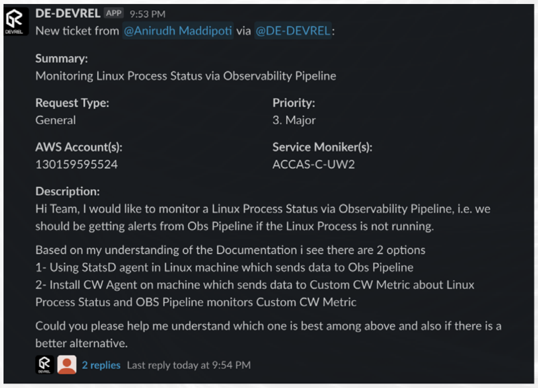
Only visible to you

DE-DEVREL APP 4:51 PM

Need Support? Click on the button below.

As soon as any query comes in through a Slack query issue, a Jira incident is created. It's automatically assigned to the dedicated support Engineer globally via load balancer.

As soon as the DevRel Engineer resolves the issue, the originator of the issue has an opportunity to accept the resolution. When the resolution is accepted, the ticket status gets updated to Closed in Jira.



DE-DEVREL APP 9:53 PM

New ticket from @Anirudh Maddipoti via @DE-DEVREL:

Summary:
Monitoring Linux Process Status via Observability Pipeline

Request Type:
General

Priority:
3. Major

AWS Account(s):
130159595524

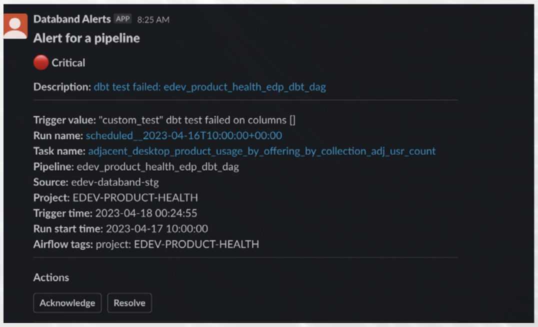
Service Moniker(s):
ACCAS-C-UW2

Description:
Hi Team, I would like to monitor a Linux Process Status via Observability Pipeline, i.e. we should be getting alerts from Obs Pipeline if the Linux Process is not running.

Based on my understanding of the Documentation i see there are 2 options
1- Using StatsD agent in Linux machine which sends data to Obs Pipeline
2- Install CW Agent on machine which sends data to Custom CW Metric about Linux Process Status and OBS Pipeline monitors Custom CW Metric

Could you please help me understand which one is best among above and also if there is a better alternative.

2 replies Last reply today at 9:54 PM



Databand Alerts APP 8:25 AM

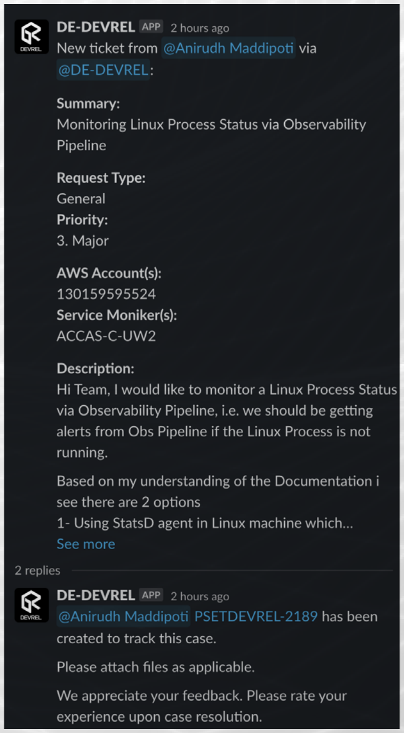
Alert for a pipeline

● Critical

Description: dbt test failed: edev_product_health_edp_dbt_dag

Trigger value: "custom_test" dbt test failed on columns []
Run name: scheduled__2023-04-16T10:00:00+00:00
Task name: adjacent_desktop_product_usage_by_offering_by_collection_adj_usr_count
Pipeline: edev_product_health_edp_dbt_dag
Source: edev-databand-stg
Project: EDEV-PRODUCT-HEALTH
Trigger time: 2023-04-18 00:24:55
Run start time: 2023-04-17 10:00:00
Airflow tags: project: EDEV-PRODUCT-HEALTH

Actions



DE-DEVREL APP 2 hours ago

New ticket from @Anirudh Maddipoti via @DE-DEVREL:

Summary:
Monitoring Linux Process Status via Observability Pipeline

Request Type:
General
Priority:
3. Major

AWS Account(s):
130159595524
Service Moniker(s):
ACCAS-C-UW2

Description:
Hi Team, I would like to monitor a Linux Process Status via Observability Pipeline, i.e. we should be getting alerts from Obs Pipeline if the Linux Process is not running.

Based on my understanding of the Documentation i see there are 2 options
1- Using StatsD agent in Linux machine which...
See more

2 replies

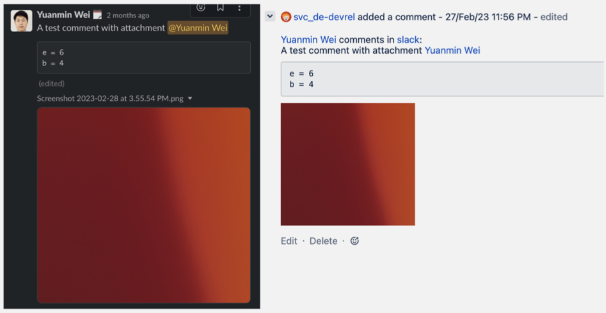
DE-DEVREL APP 2 hours ago

@Anirudh Maddipoti PSETDEVREL-2189 has been created to track this case.

Please attach files as applicable.

We appreciate your feedback. Please rate your experience upon case resolution.

- Data sync:
  - Name tagging
  - URLs
  - Code blocks
  - Attachments
- Link to original Slack thread available



Yuanmin Wei 2 months ago
A test comment with attachment @Yuanmin Wei

```
e = 6
b = 4
```

(edited)
Screenshot 2023-02-28 at 3.55.54 PM.png

svc_de-devrel added a comment - 27/Feb/23 11:56 PM - edited

Yuanmin Wei comments in slack:
A test comment with attachment Yuanmin Wei

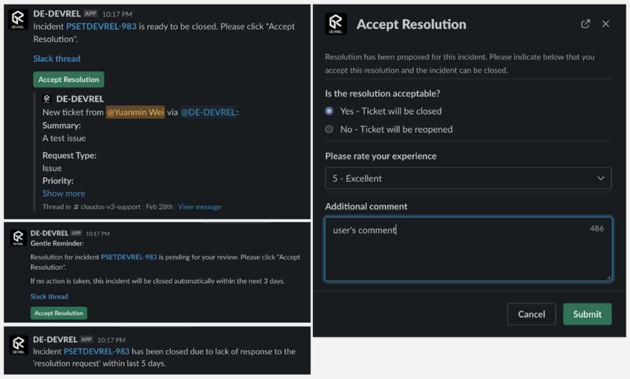
```
e = 6
b = 4
```

We collect our ratings, and feedback is welcome. If the resolution is not accepted the Jira incident immediately gets reverted to the in-progress state and gets reassigned to the DevRel engineer to take a further look.

The solution is fully customizable. Meaning platform teams can specify the details developers can enter based on their technology. The Continuous Integration/ Continuous Delivery (CI/CD) team can ask developers for details related to Jenkins-related incidents, accelerating the time to resolution.

Feature – CSAT Collection

- A ”resolution request” via Slack DM is sent to user upon Jira issue resolution
  - User decides “yes” or “no” with a rating and feedback comment
- A reminder message is sent after 2 days if user does not respond
- Jira issue is auto-closed after 5 days if user does not respond



DE-DEVREL APP 10:17 PM
Incident PSETDEVREL-983 is ready to be closed. Please click "Accept Resolution".

Slack thread

DE-DEVREL
New ticket from @Yuanmin Wei via @DE-DEVREL:
Summary:
A test issue
Request Type:
Issue
Priority:
Show more
Thread in # cloudos-v3-support | Feb 28th | View message

DE-DEVREL APP 10:17 PM
Gentle Reminder:
Resolution for incident PSETDEVREL-983 is pending for your review. Please click "Accept Resolution".
If no action is taken, this incident will be closed automatically within the next 3 days.
Slack thread

DE-DEVREL APP 10:17 PM
Incident PSETDEVREL-983 has been closed due to lack of response to the 'resolution request' within last 5 days.

Accept Resolution

Resolution has been proposed for this incident. Please indicate below that you accept this resolution and the incident can be closed.

Is the resolution acceptable?
● Yes - Ticket will be closed
○ No - Ticket will be reopened

Please rate your experience
5 - Excellent

Additional comment
user's comment 486

NATURAL LANGUAGE

Onboarding Steps

Preparation

- Partner team prepares the needful:
  - UI workflows details
  - Slack channel names, Jira project, issue type, fields
  - Add DE-DEVREL’s centralized Jira service account to Jira project
- Partner team meets with DE-DEVREL for onboarding details

Integration

- DE-DEVREL does the implementation
- DE-DEVREL presents a demo implementation to partner team
- DE-DEVREL incorporates feedback and aligns with partner team on production rollout date

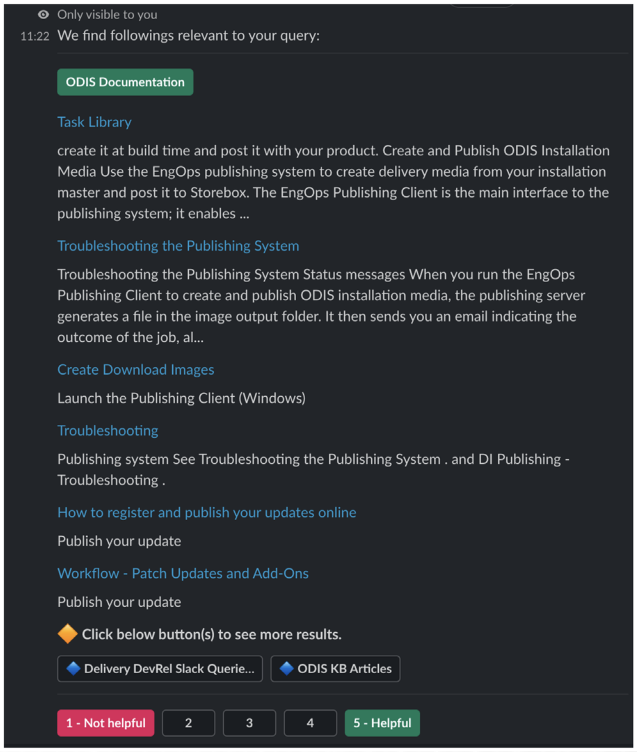
Deployment

- DE-DEVREL deploys the integration to production
- DE-DEVREL shares following with partner team:
  - A user guide
  - Analytics dashboard access secret

PROCESSING AND AI

We extended the capabilities of the platform by launching a Self-Help AI Bot. An AI bot that taps into all content management spaces and those content management spaces can be anything from platform documentation. Users are able to receive immediate answers and prioritize any queries they have without the need for manual intervention.

We used natural language processing to serve knowledge-based articles for resolving issues of a similar nature. Reusing support engineering resolutions effectively reduces the amount of repeated work. All developers in the ecosystem can learn from one another.



Only visible to you

11:22 We find followings relevant to your query:

ODIS Documentation

Task Library

create it at build time and post it with your product. Create and Publish ODIS Installation Media Use the EngOps publishing system to create delivery media from your installation master and post it to Storebox. The EngOps Publishing Client is the main interface to the publishing system; it enables ...

Troubleshooting the Publishing System

Troubleshooting the Publishing System Status messages When you run the EngOps Publishing Client to create and publish ODIS installation media, the publishing server generates a file in the image output folder. It then sends you an email indicating the outcome of the job, al...

Create Download Images

Launch the Publishing Client (Windows)

Troubleshooting

Publishing system See Troubleshooting the Publishing System . and DI Publishing - Troubleshooting .

How to register and publish your updates online

Publish your update

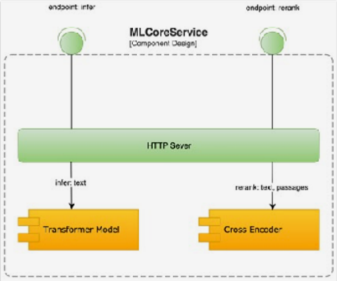
Workflow - Patch Updates and Add-Ons

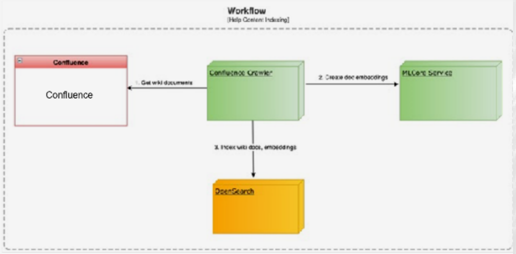
Publish your update

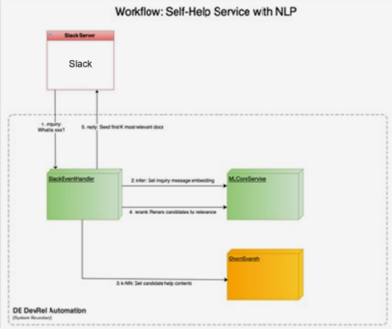
🔶 Click below button(s) to see more results.

Self-Help Service with NLP

- MLCore Service:
  - An HTTP server with NLP transformer model loaded providing endpoints for inference and reranking
  - Inference API takes in a sentence and calls transformer to produce a sentence embedding
  - Rerank API takes a sentence, a list of passages to produce similarity values for each (sentence, passage) pair
- Help Content Processing:
  - Help content needs be indexed with an embedding format
  - Document crawler is a web crawler that crawls web page contents or a script that accesses documents on content server via SDK
- Search & Integration



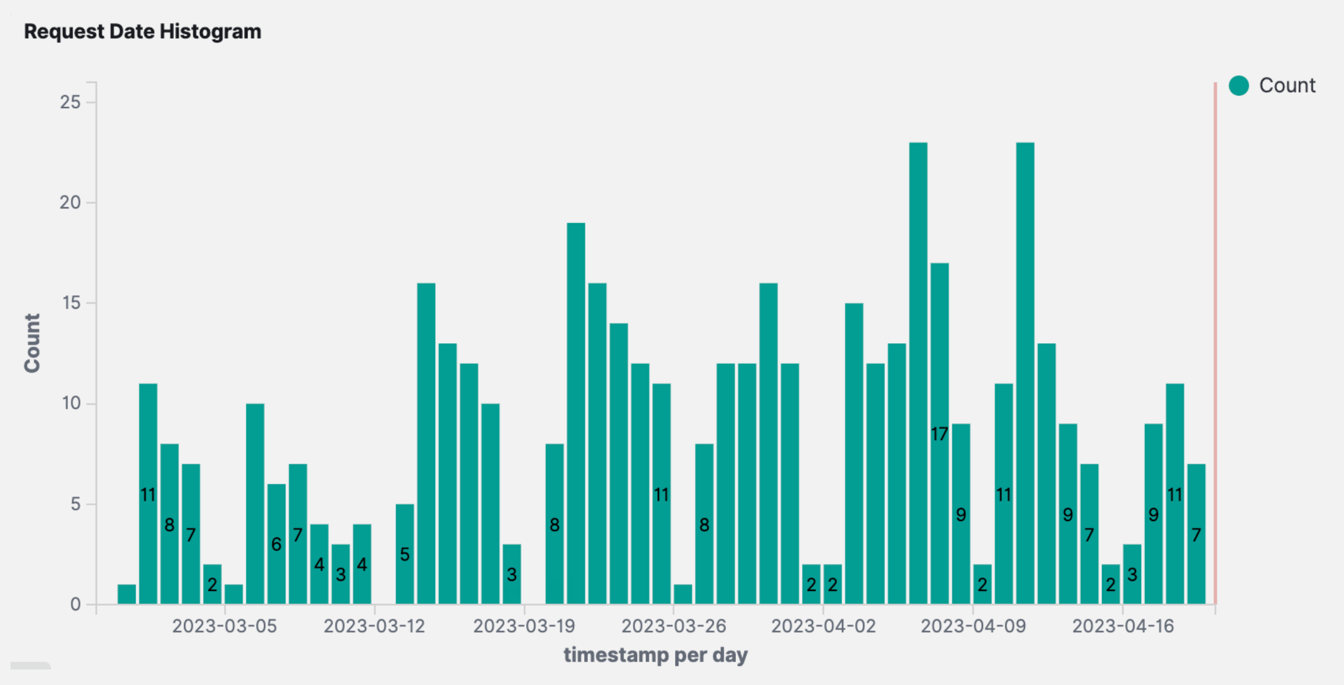


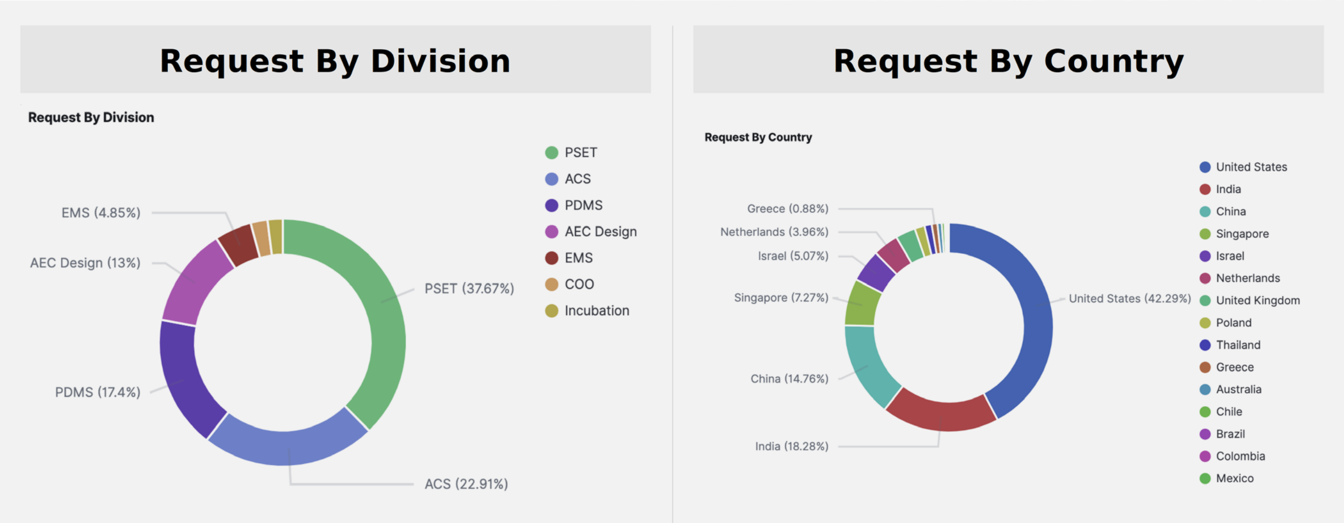

Workflow: Self-Help Service with NLP
Slack Server
Slack
1. inquiry: What is xxx?
5. reply: Send first K most relevant docs
SlackEventHandler
2. infer: Get inquiry message embedding
4. rerank: Rerank candidates by relevance
MLCoreService
OpenSearch
3. k-NN: Get candidate help contents
DE DevRel Automation
[System Boundary]

All developers in the ecosystem can learn from one another.

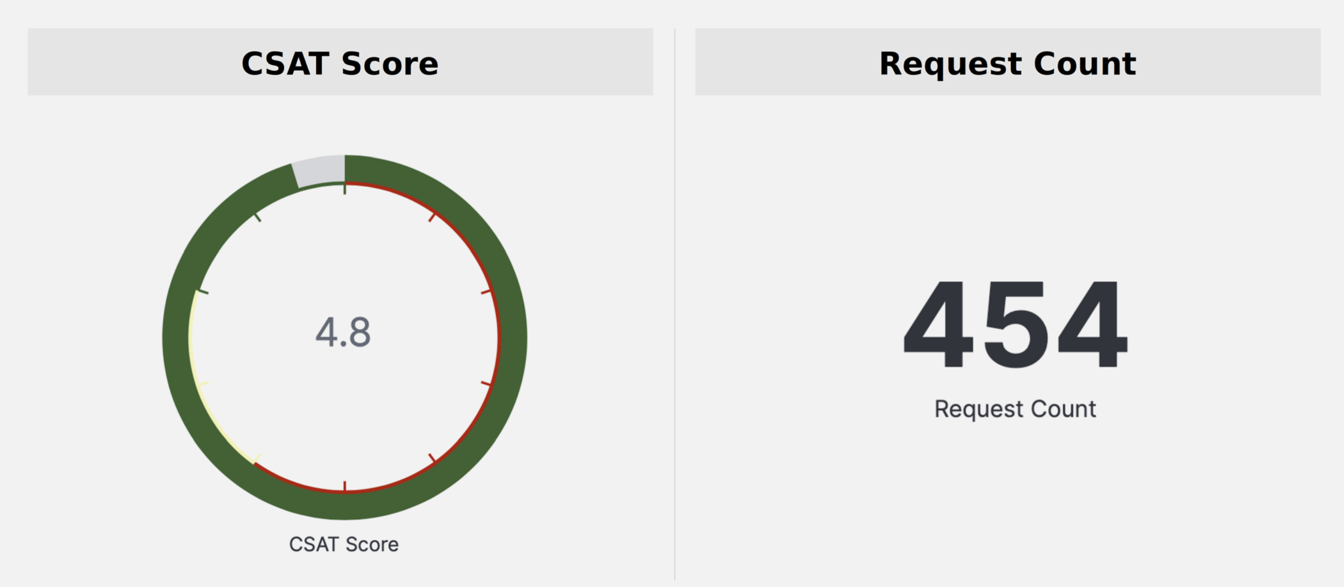
ANALYTICS & REPORTING

We created functionality to generate dashboards and reporting based on relevant organizational filters and specifications. Every team is given the necessary credentials to generate relevant reports on ticket resolutions and solutions.





IMPACT



CSAT Score

4.8

CSAT Score

Request Count

454

Request Count

The largest challenge in building solutions for better software delivery is maintaining and building customer trust. Once a customer's mind is made up, it's extremely hard to convince them of other solutions to resolve their problem. We built a DevRel Automation Platform to boost developer productivity, experience, and trust—this then led us to greater customer satisfaction. Today, we have an average Customer Satisfaction Score (CSAT) of 4.8 across delivery partners for CI/CD, Git, and Observability.

By creating a repeatable and sustainable framework for software delivery, we improved the mean time to restore, lead time for changes, incident resolution time, code and test review turnaround time, and number of platform feature releases. This resulted in greater trust within Autodesk and improved customer satisfaction.

CONCLUSION

Our Autodesk DevRel Automation Platform explores new ways of integrating open source knowledge with Continuous Delivery practices that promote productivity and developer trust. The streamlined development process and enhanced collaboration have sparked interest in exploring new market segments and diversifying feature offerings. Developers have many workflows that involve tools that could benefit from automation and integration, such as ServiceNow, PagerDuty, Salesforce, etc., and other commonly utilized delivery technologies. Security, compliance, and cost savings are also at the forefront of Continuous Delivery, and we look forward to integrating practices that build these values into our delivery.

We built a DevRel Automation Platform to boost developer productivity, experience, and trust—this then led us to greater customer satisfaction.

CD.FOUNDATION