

servicenow®

Creating a **Service Aware** Configuration Management Database

Built for Cloud, Optimized for Service

- **Mike Buckner**, IT Operations Management Solutions Architect
- **ServiceNow** | The Enterprise Cloud Company

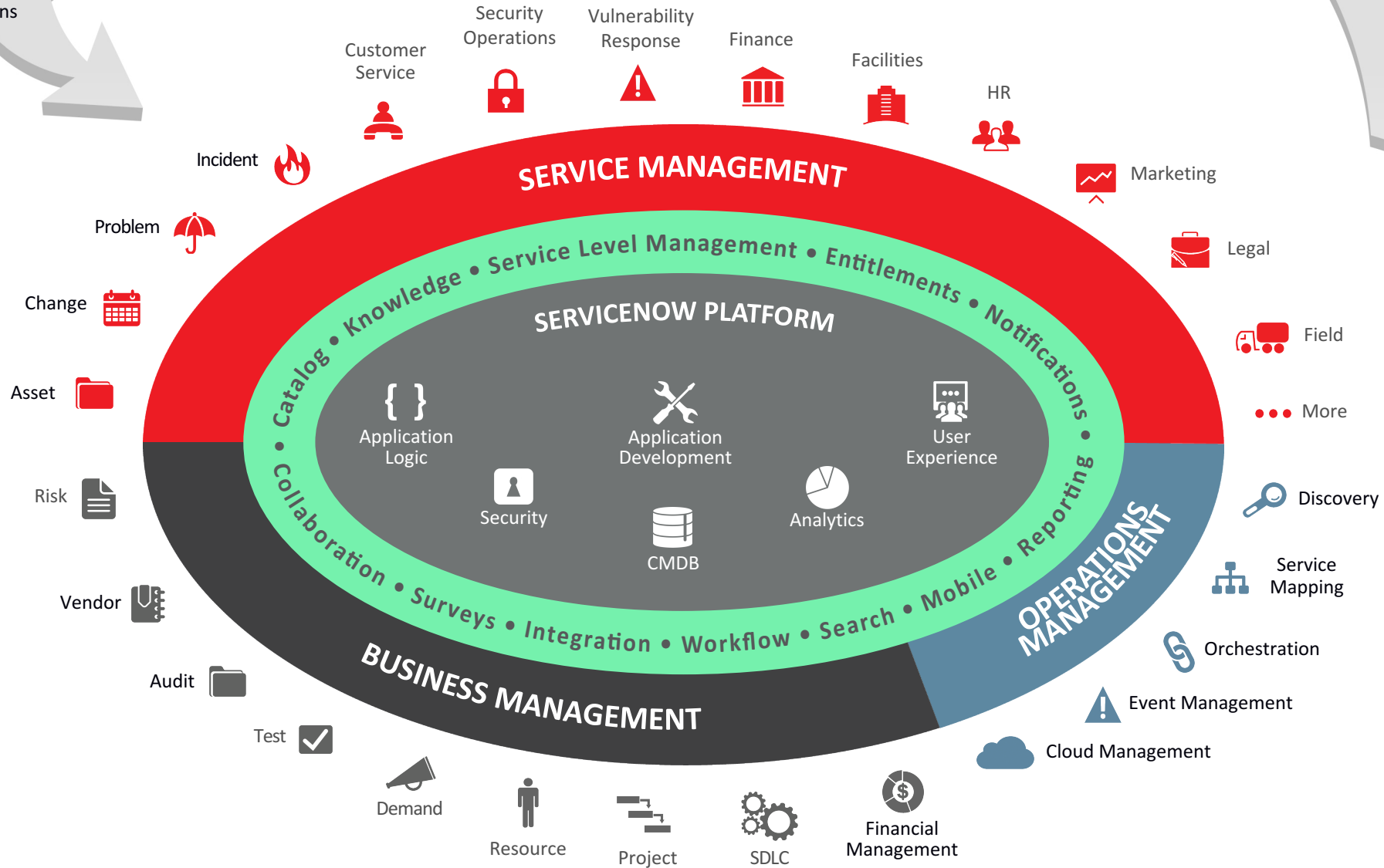




Service Mapping & Discovery Overview

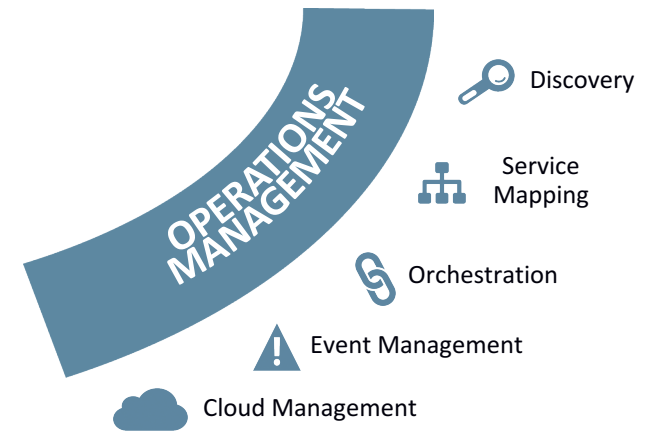
Secure Integrations

Secure Integrations



Secure Integrations

Secure Integrations

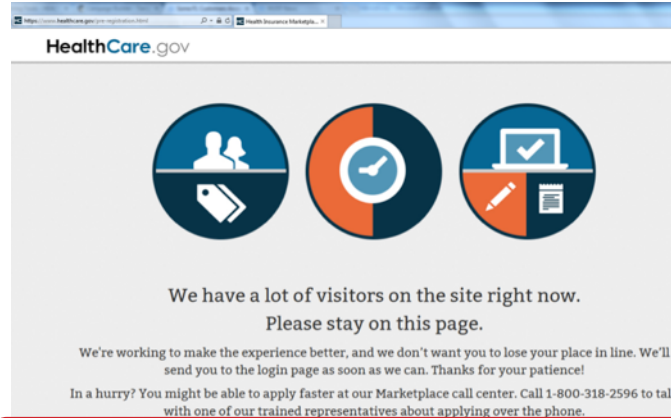


Services Top of Mind with CIOs Everywhere



Visibility 

“I can’t tell which systems are connected to each service”



Availability 

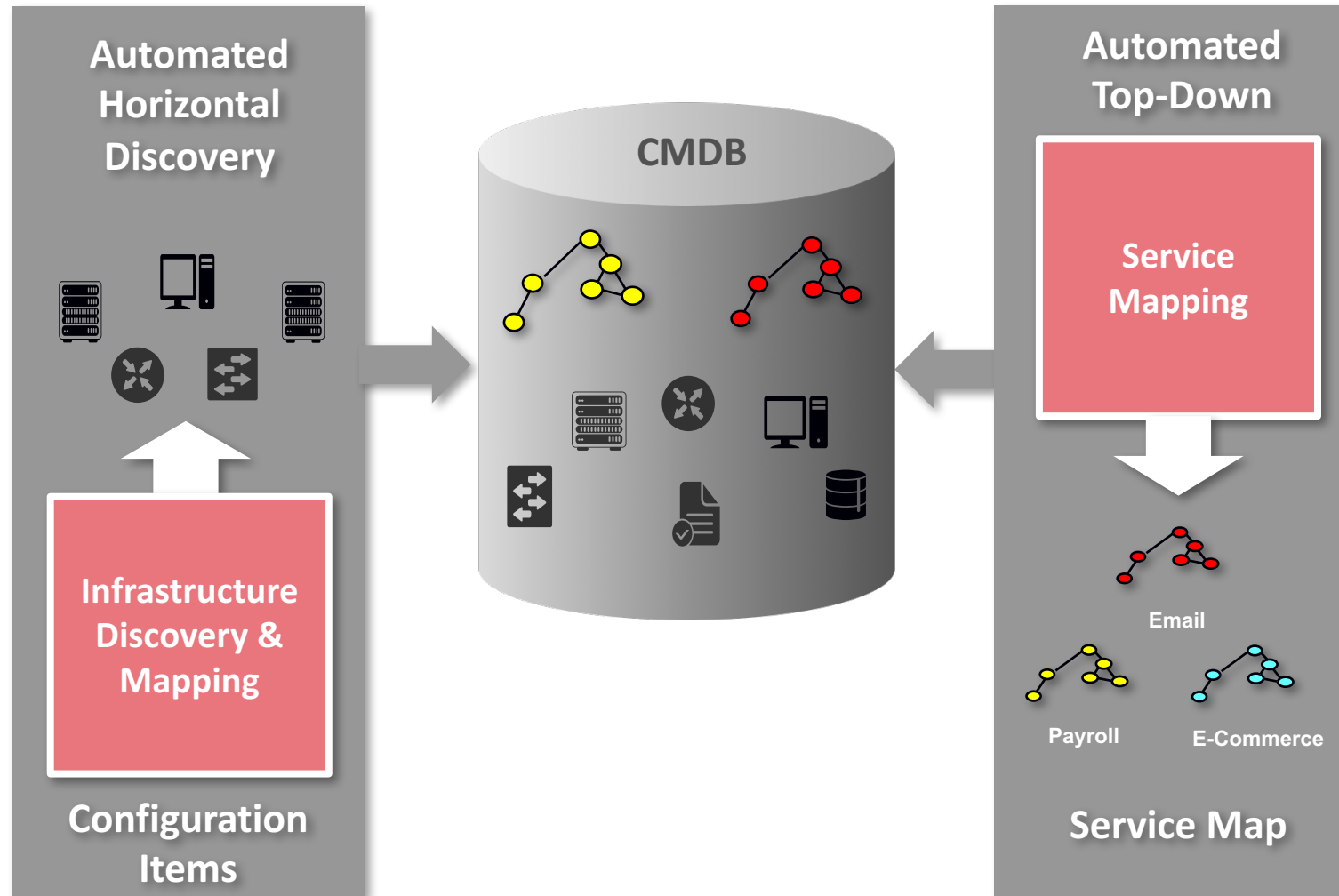
“If a service goes down, I can’t find the cause of the outage”



Agility 

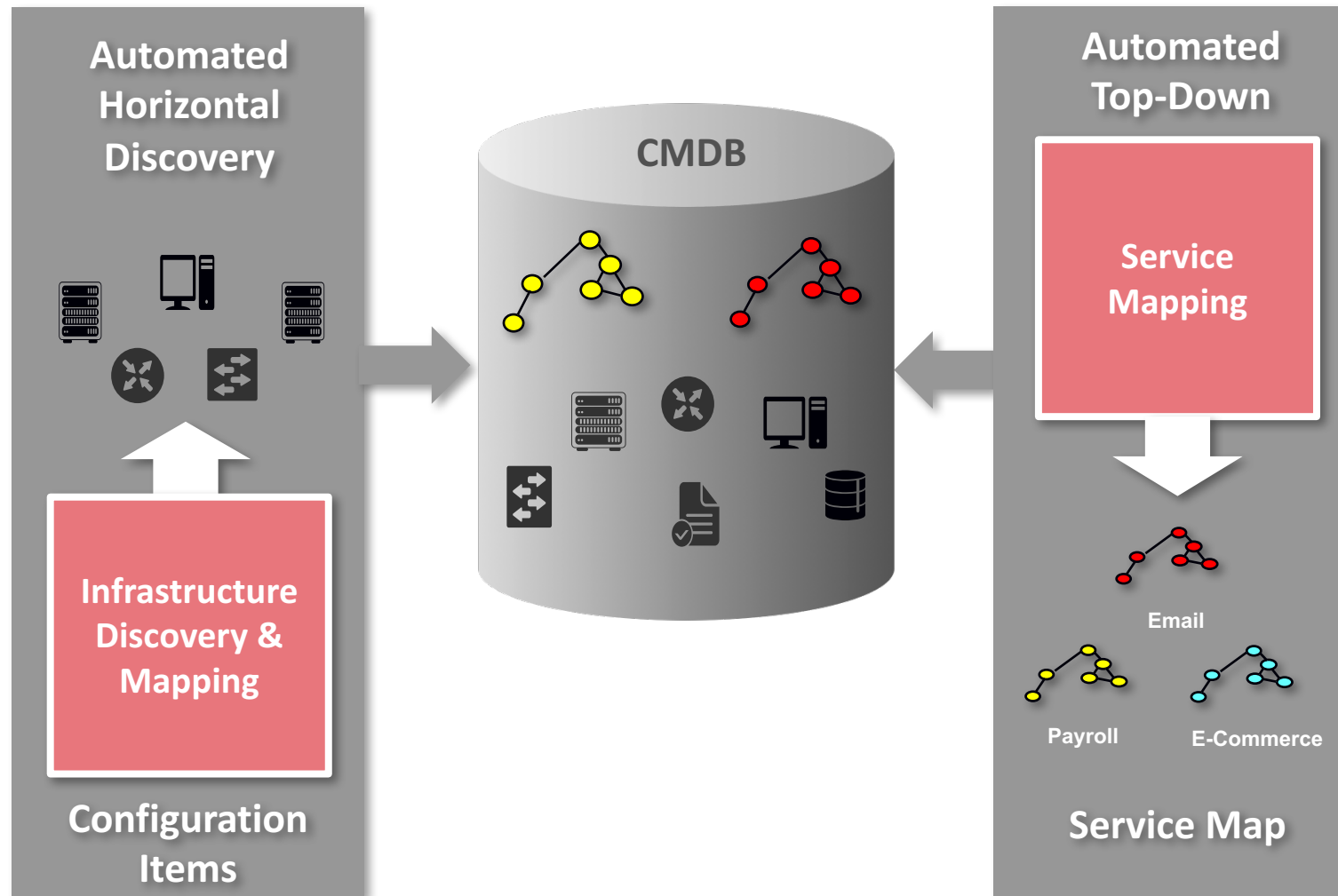
“We need to accelerate service delivery, while maintaining quality”

Creating a Service Aware CMDB



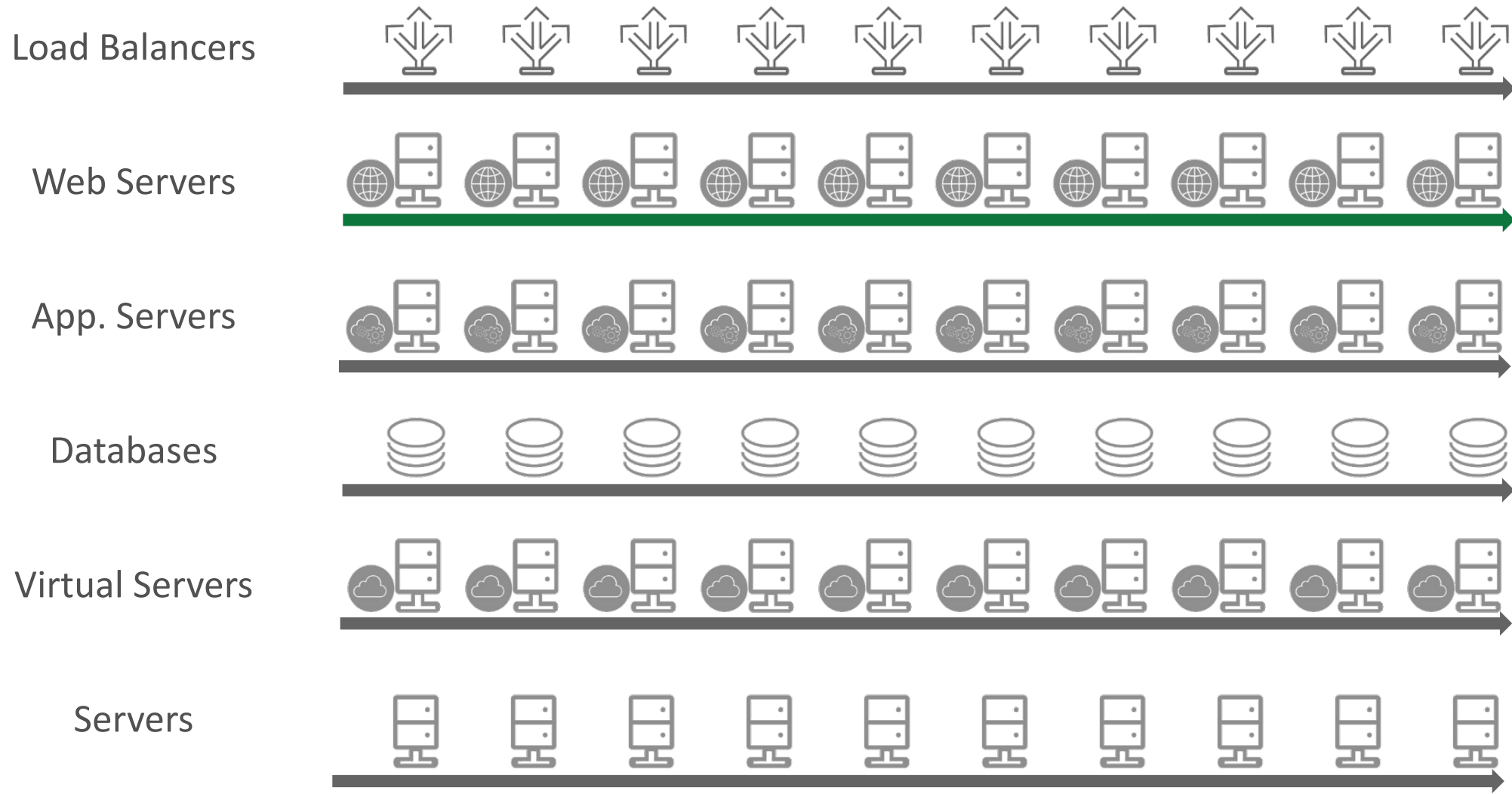
- Populates the CMDB with the current state of the environment
- Pulls in detailed CI information about all of your IT assets
- Accommodates “real time” state changes for dynamic environments such as VMware or AWS
- Complements service relationships from service mapping

Creating a Service Aware CMDB

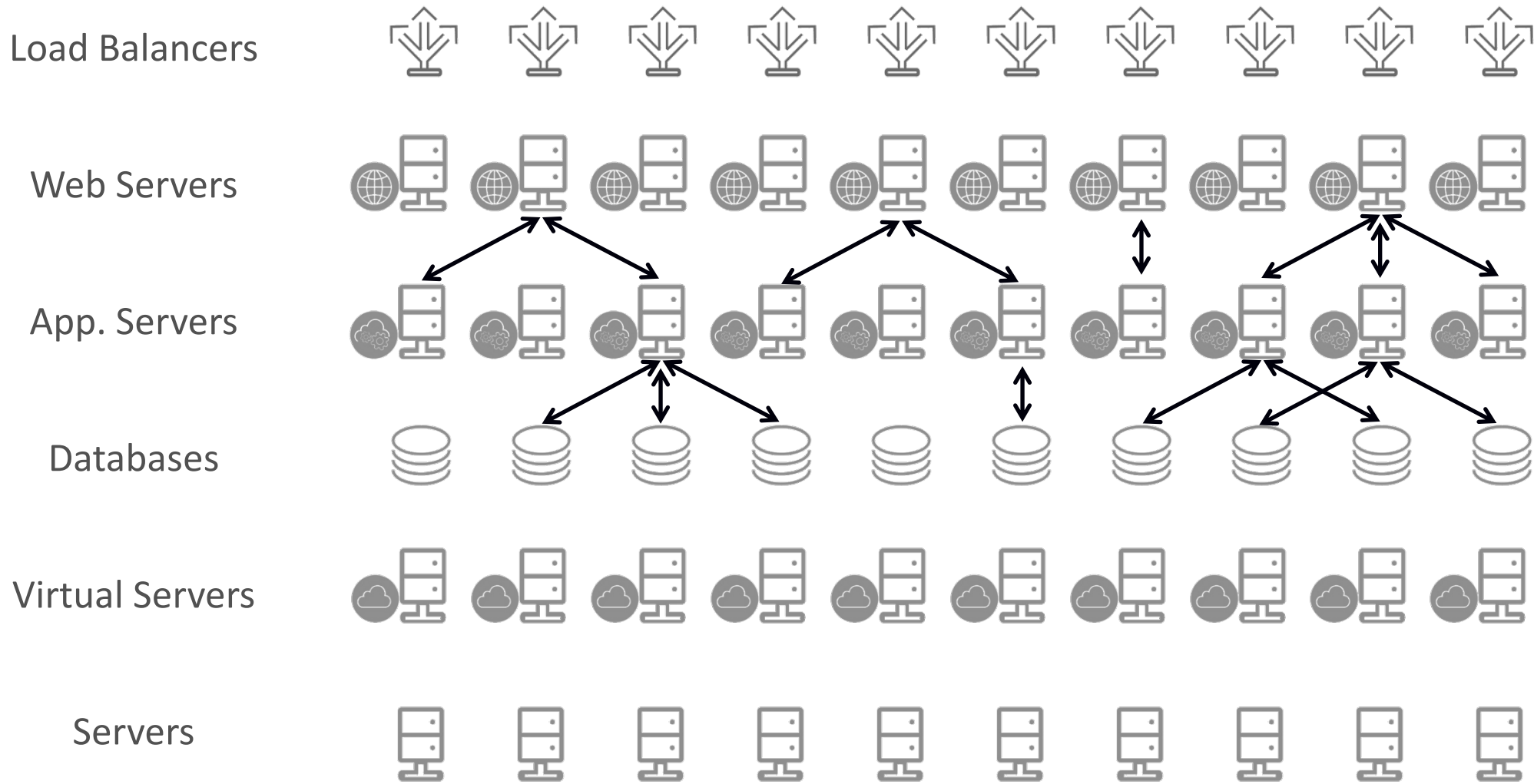


- Populates the the CMDB with complex relationships between CIs automatically
- Builds network (L2 & L3) relationships between CIs
- Automatically discovers application architecture constructs such as load balancing and clusters
- Creates a tight service-centric context to the CIs in your CMDB

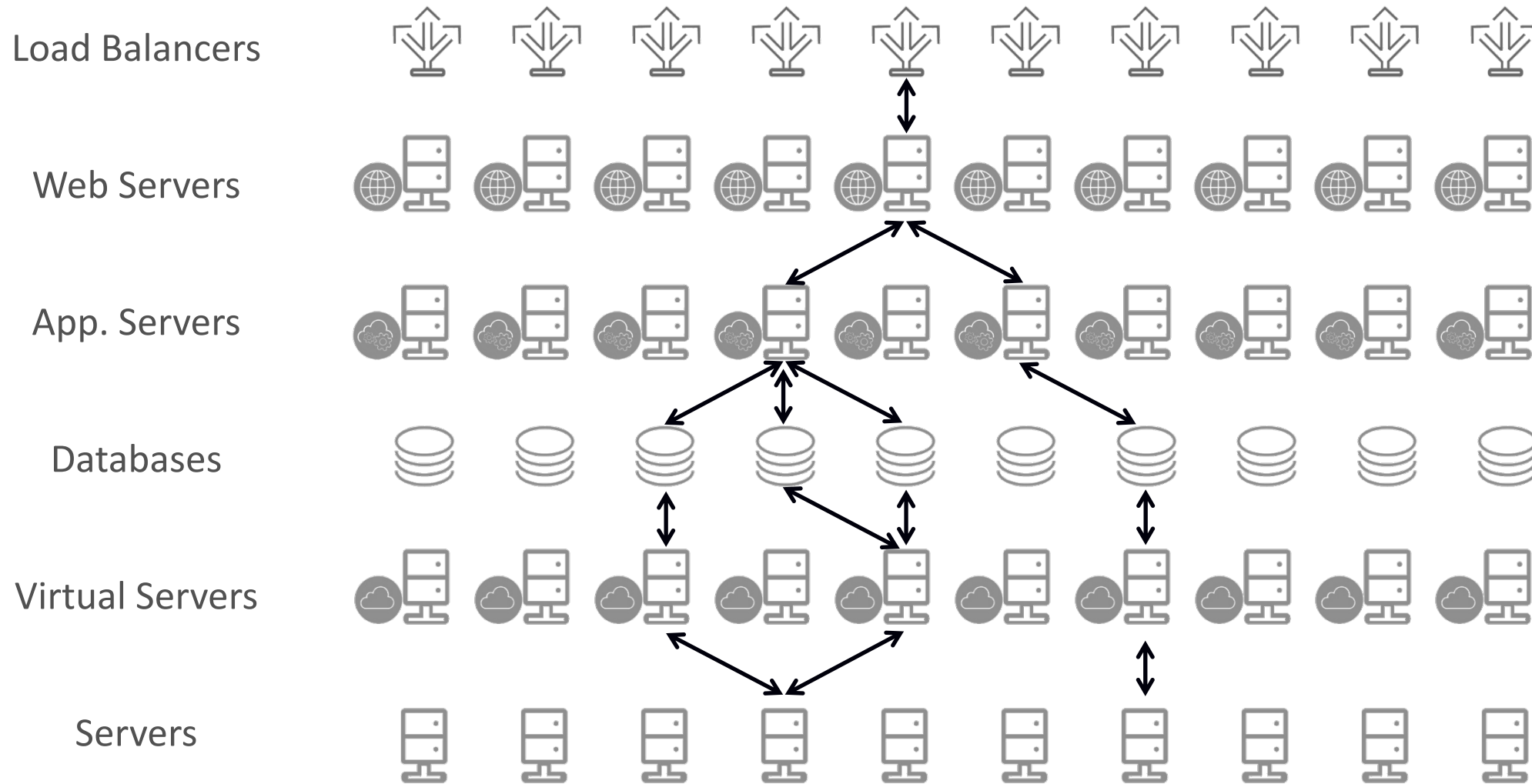
Traditional Infrastructure Discovery



Application Dependency Mapping (Traditional Discovery cont.)

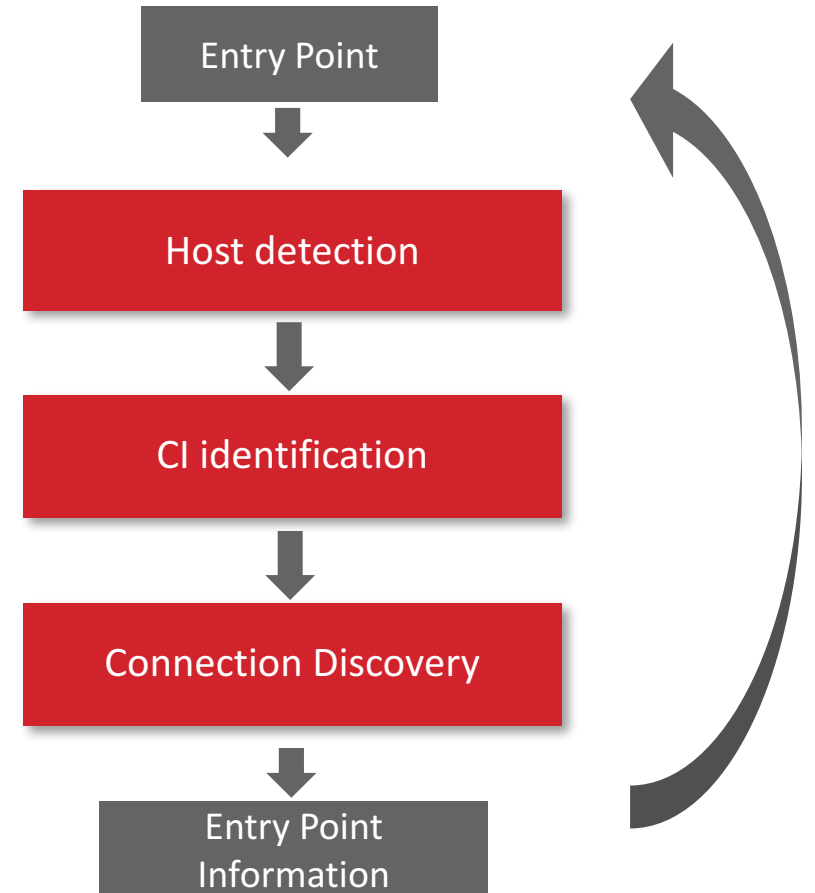


Service Mapping Provides True Business Context



The Service Mapping Process

- **Entry Point**
 - URL, connection parameters, etc.
- **Host detection**
 - Connection to target machine, discovery CI information
- **CI Identification**
 - Identify the application based on information from entry point
- **Connection Discovery**
 - Discover configured connection to other applications



Pattern Based Approach

- Service Mapping accomplishes this by using a simple pattern based approach to mapping.
- Patterns are broken up into two sections:
 1. Identification – This is where we look for the application signature. It can be as simple as checking if the process bound to the port associated with our entry point is an expected name or value (e.g. Apache or Tomcat).
 2. Connectivity – This is where we look at configuration files, persistent network connections, or any other forms of metadata to understand what our downstream dependencies are.

The screenshot shows the configuration interface for a .NET Application. The window is titled ".NET Application" and is divided into three main sections: Definition, Identification Sections, and Connectivity Sections.

Definition: This section contains several input fields and dropdown menus. The "Name" field is populated with ".NET Application". The "Description" field is empty. The "Operating system" dropdown is set to "Windows:All". The "Run order" dropdown is set to "After", and the "IIS" dropdown is set to "IIS".

Identification Sections: This section contains a list of identification rules. The first rule is "Identification for HTTP(S), TCP". The second rule is "Identification for remote TCP".

Connectivity Sections: This section contains a list of connectivity rules. The first rule is ".NET remoting connectivity". The second rule is "XML Web Services connectivity". The third rule is "ADO .NET connectivity".

At the bottom of the window, there are four buttons: "Save", "Activate", "Cancel", and "Check Pattern".

Pattern Based Approach

- Inside each of these sections we use the Pattern Designer, a simple WYSIWYG editor to build patterns to look for our application signature, or to parse configuration files to understand dependencies.
- No programming required, just an understanding of Windows, Linux, and SNMP commands.

The screenshot shows the Pattern Designer interface for a pattern named "Oracle DB On Unix Pattern". The main window is titled "Identification for Oracle DB entry point type(s) - no instance". The interface includes a left sidebar with a list of steps, a central workspace for defining operations, and a right sidebar for managing variables and attributes.

Oracle DB On Unix Pattern

- 1. match we didnt get instance name
- 2. Check process name to match Ora
- 3. ps pmton
- 4. find the pmton process
- 5. match
- 6. if dynamic config - make sure we fo
- 7. extract ora home
- 8. set ora_home_exe
- 9. set process
- 10. export oracle home
- 11. if dynamic config - take the instar
- 12. take the ora home from listener p
- 13. sets the display label
- 14. Get version
- 15. Get version from path
- 16. Reference to library with id=25FAE
- 17. Check Mandatory Attributes

Identification for Oracle DB entry point type(s) - no instance

Operation: Match

Set Precondition:

\$process.commandLine Contains "tnslsnr"

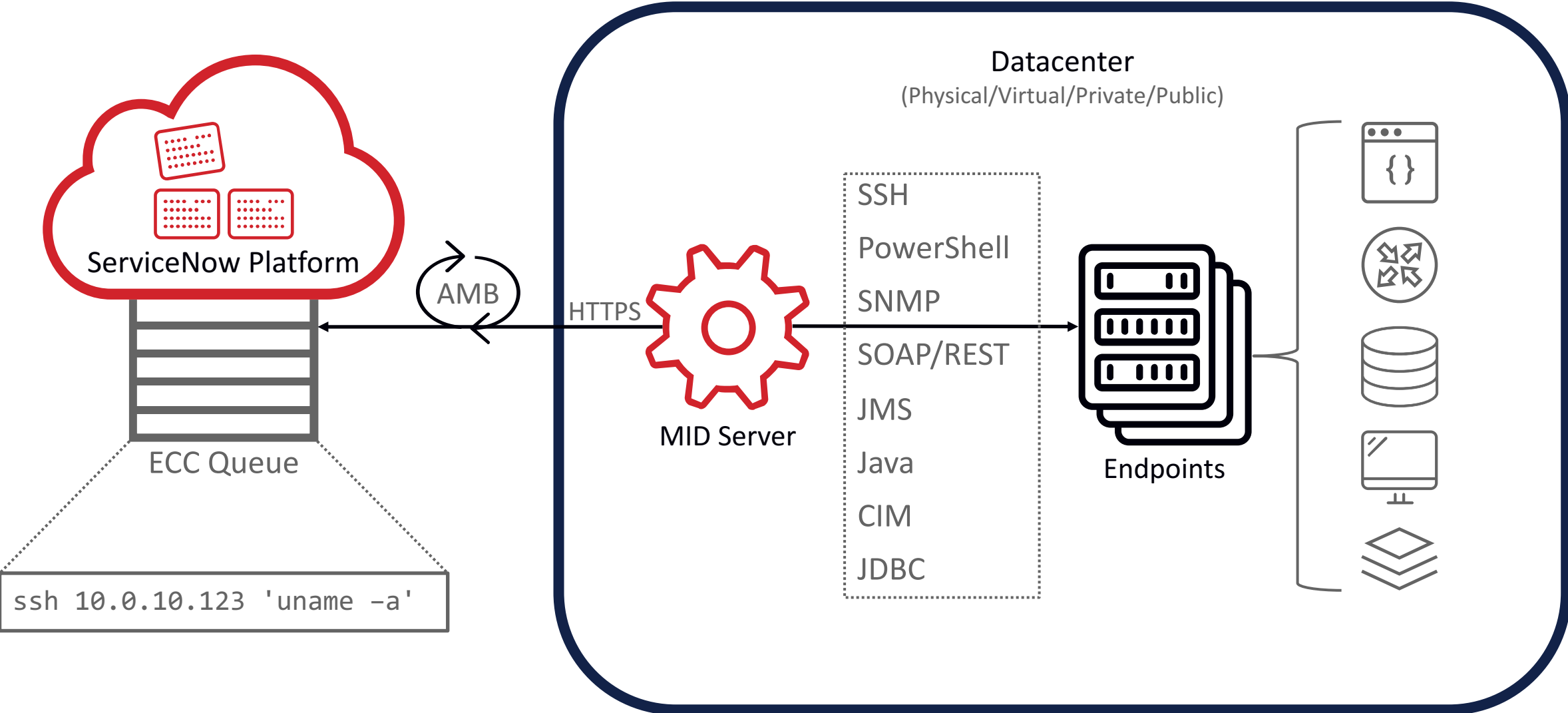
Temporary Variables

Name	Value
computer_system	
process	

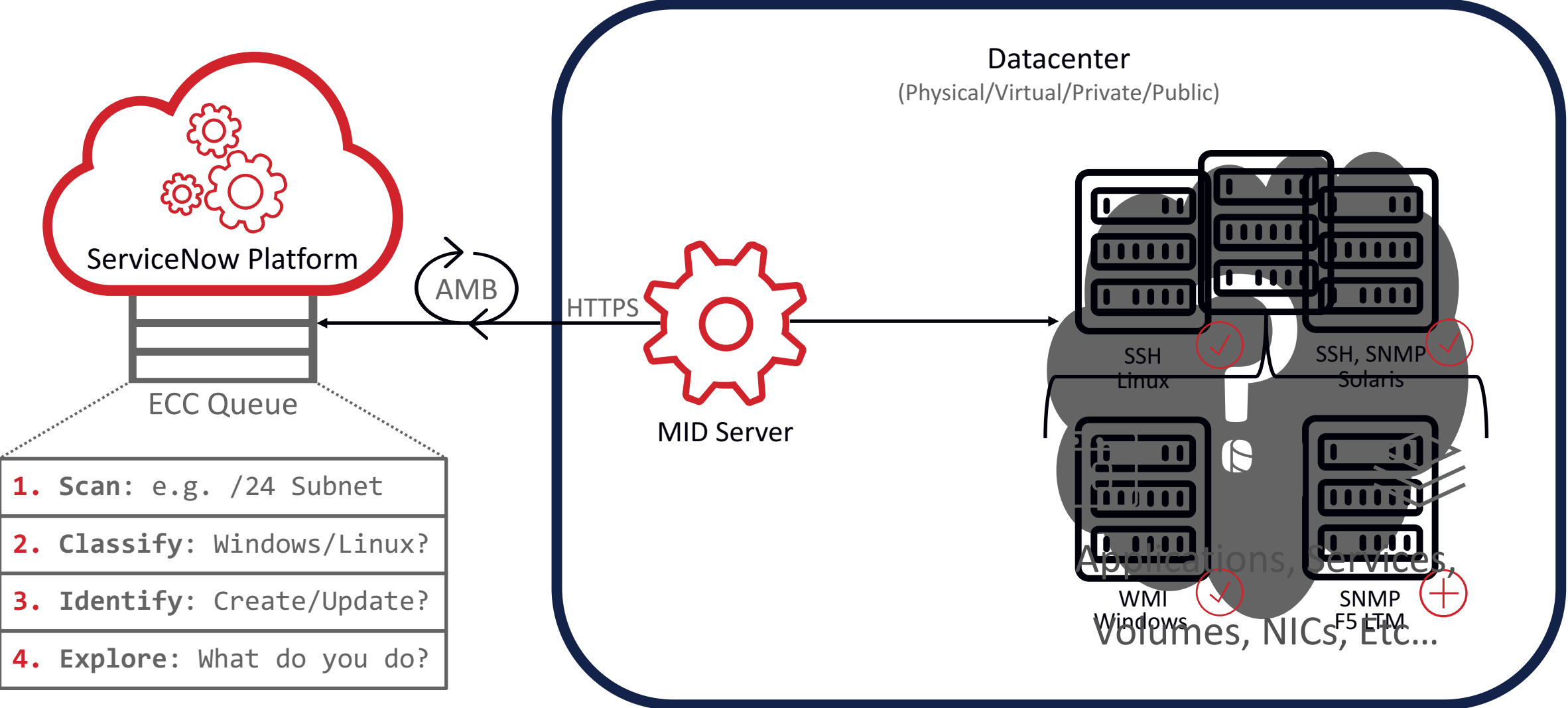
CI Attributes

Name	Value
install_directory	
running_process_comm	
version	
listener_name	
config_file	
edition	
name	
used_for	
config_directory	
running_process	
tcp_port	
pid	
sid	

MID Server Architecture

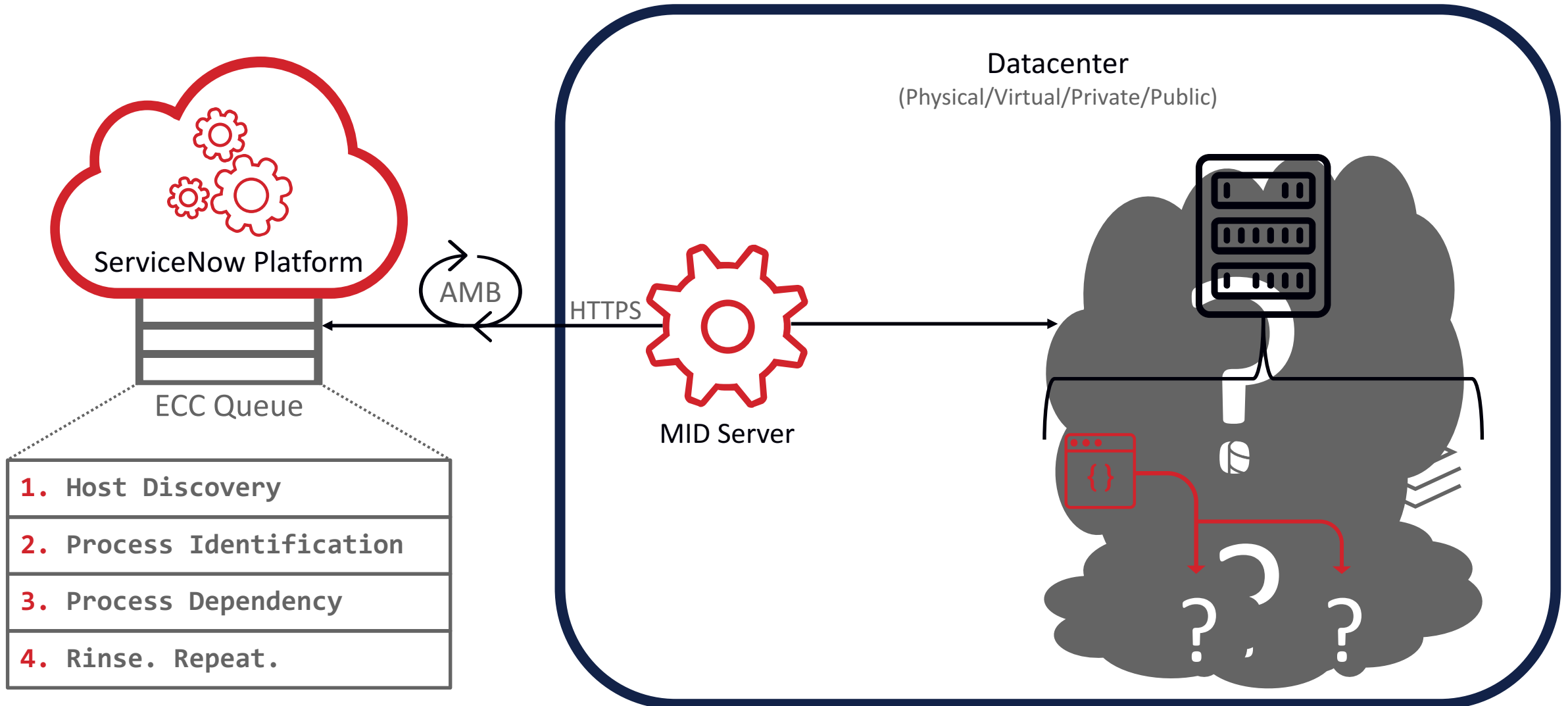


Host Discovery Process



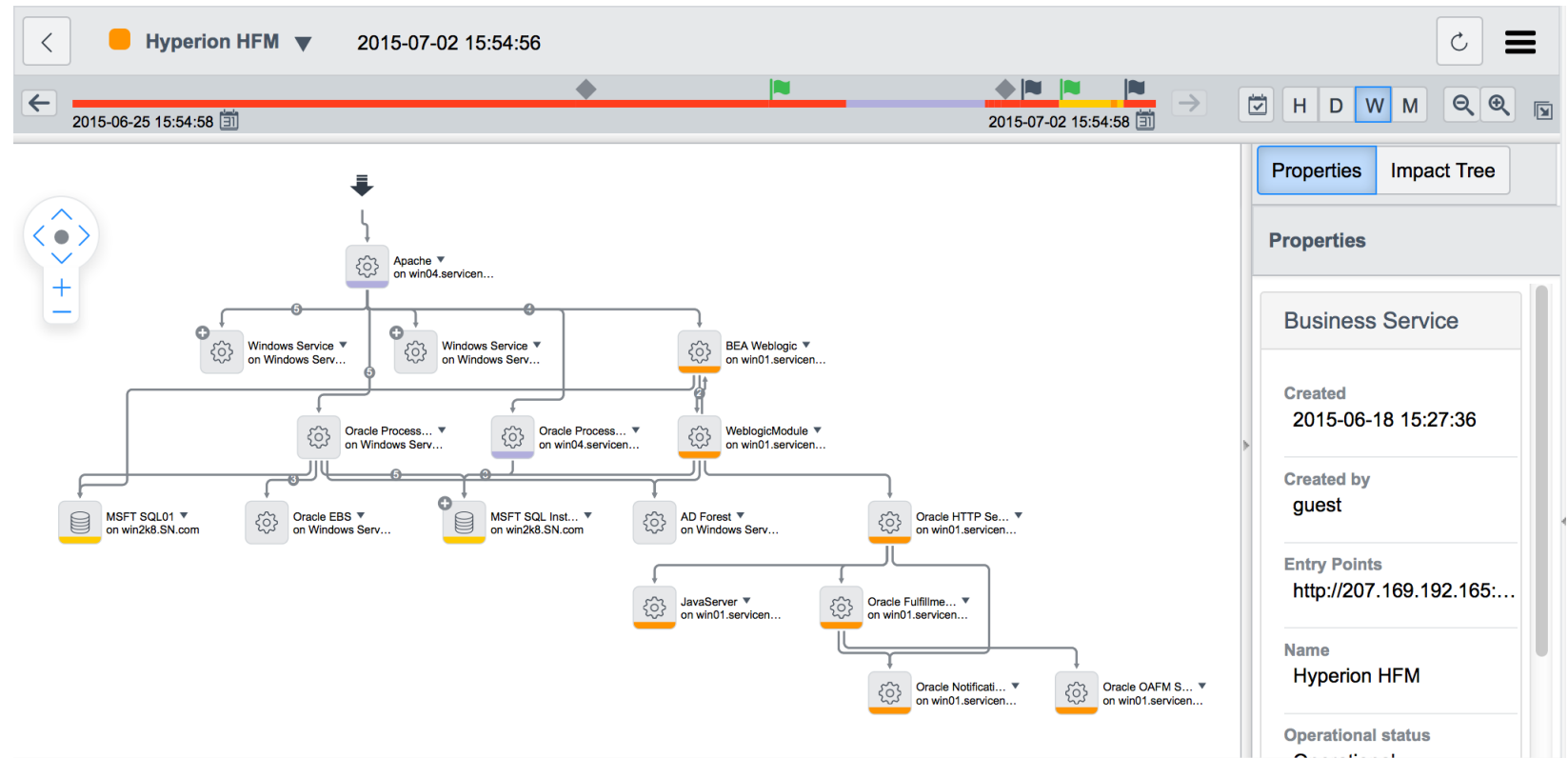
- 1. Scan: e.g. /24 Subnet
- 2. Classify: Windows/Linux?
- 3. Identify: Create/Update?
- 4. Explore: What do you do?

Service Discovery Process



Service Mapping Outcomes

- **Automatically update** service maps so data is current and accurate
- Instantly identify **impact of planned changes** to help prevent issues
- Identify unplanned changes to **ensure compliance**
- Pinpoint root cause of service disruption to **speed time to resolution**

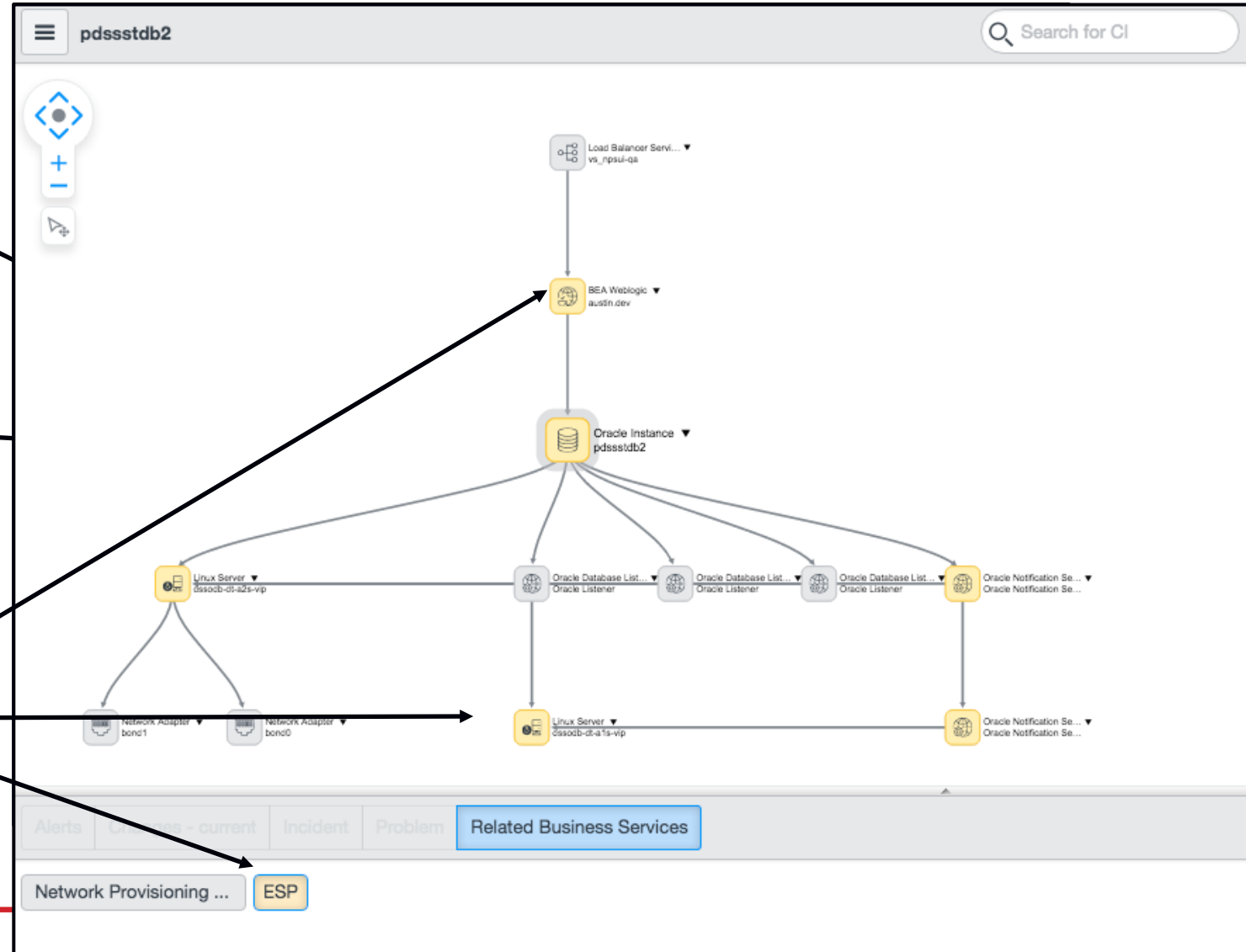




Service-Aware CMDB Use Cases

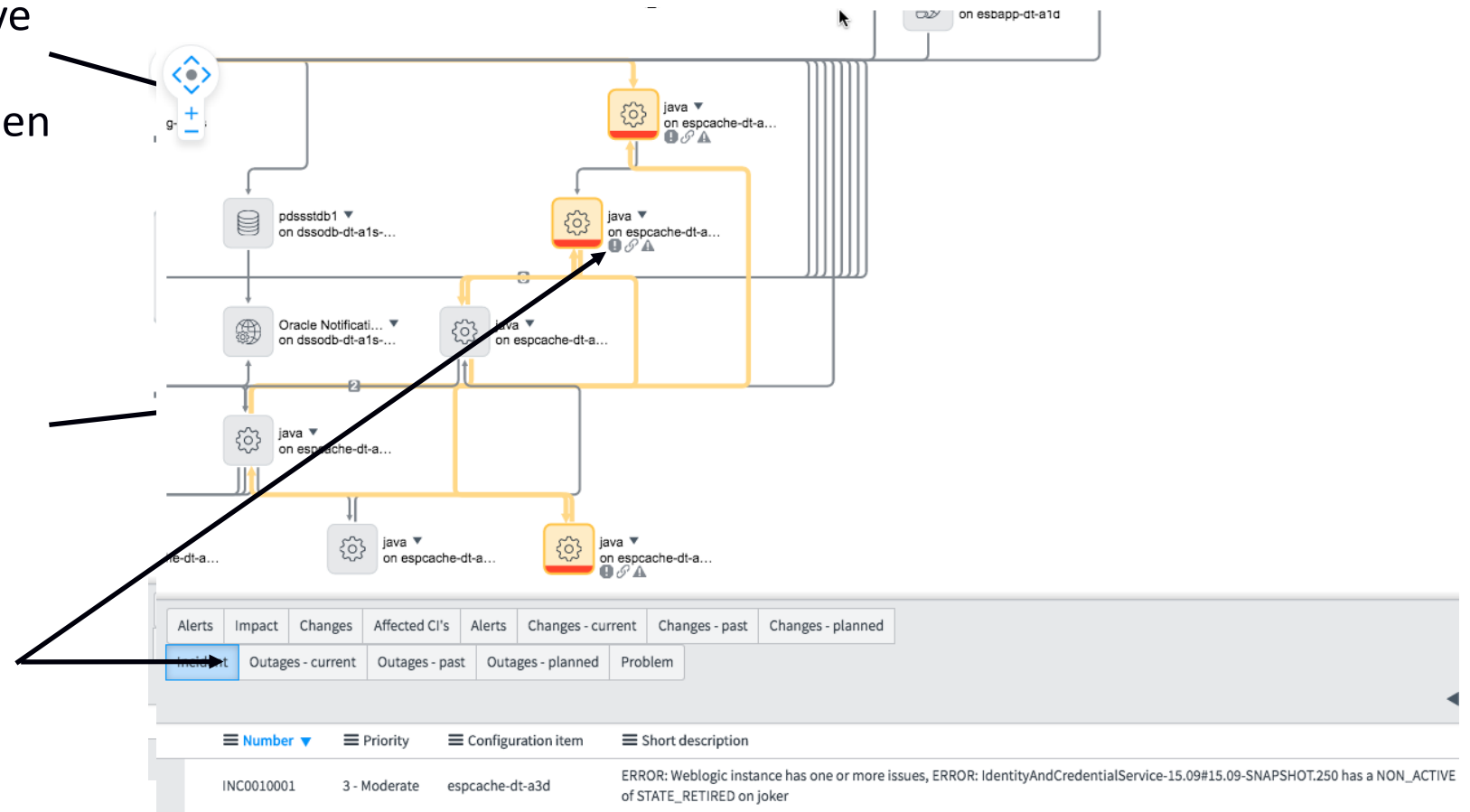
1. Business Service – Incident & Change Impact Analysis

- For any change request for the server “pdsstdb2”
- click on the button to see the relationships between the objects.
- graphically the server and its relationship to configuration items and the names of Related Business Services.



2. Business Service – Detect Changes & Incidents

- Ability to see which CIs have approved change requests which are planned to happen in the future
- Identify “unauthorized” or “Unplanned” Changes”
- Integration of logged Incidents to CI’s



3. Business Service – Historical Comparison

- Compare state of the Business Service over time

- Identify which CI's has changed

- Specific details of CI changes (example data)

The screenshot displays the ServiceNow Business Service Historical Comparison interface. It shows a comparison between two baselines: 'Set baseline' (2016-08-05 08:50) and 'Point 1: 2016-08-10 11:22'. The main view shows a diagram of components. A component labeled 'pdssstdb2 on dssodb-dt-a2s-...' is highlighted with a blue box and labeled 'UPDATED'. A 'Properties' panel is open for this component, showing the following details:

Property	Value
Serial number	USE43594WN
Cluster node count	0
CI Type	Linux Server
Class	cmdb_ci_linux_server
Kernel Release	2.6.32-358.23.2.el6.x86_64
RAM (MB)	Updated Old: 96705 New: 106000
Host related items	Network Adapter (2) Serial Number (3)

4. Service Mapping + Event Management = Impact Analysis

- Dive into the Business Service by “clicking” on red box.

- Visually see specific server on which the event exist from Nagios, etc..

- Specific details on what triggered alert/event

The screenshot displays a ServiceNow interface. The top portion shows a service map with various components and their dependencies. A red box highlights a specific component, and an arrow points from the first bullet point to it. Below the map, a table lists alerts. An arrow from the third bullet point points to the 'Description' column of the alert table.

Number	Type	Severity	Description	Source	Configuration item	Node	Task
Alert0010005	WLS Health Check for Joker 8311	Critical	ERROR: Weblogic instance has one or mor...	Nagios	espcache-dt-a3d	espcache-dt-a3d.downingtown.pa.ula.comca...	INC0010001
Alert0010006	WLS Health Check for Joker 8411	Critical	ERROR: Weblogic instance has one or mor...	Nagios	espcache-dt-a3d	espcache-dt-a3d.downingtown.pa.ula.comca...	
Alert0010007	WLS Health Check for Joker	Critical	ERROR: Weblogic instance has one or mor...	Nagios	espcache-dt-a3d	espcache-dt-	

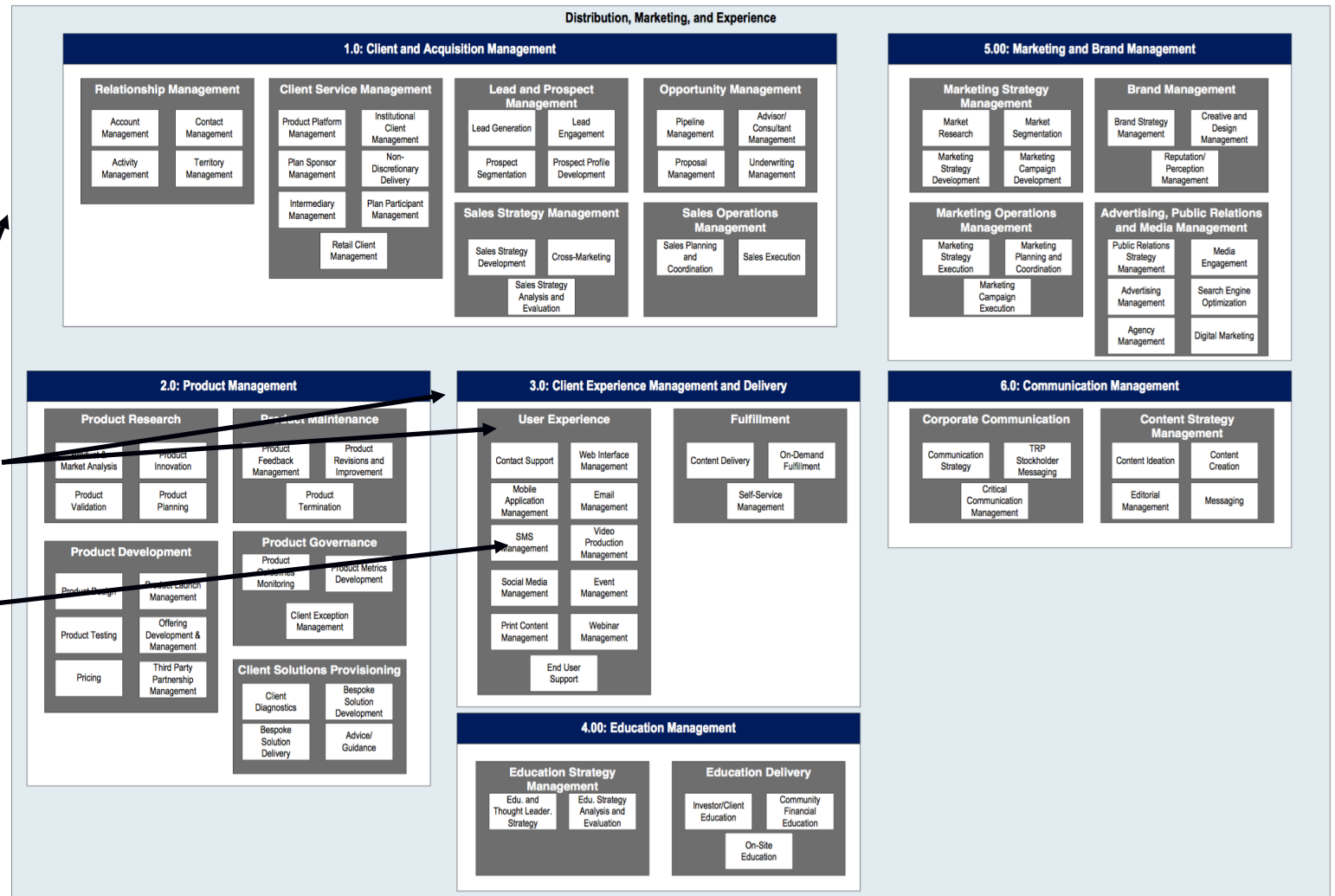
5. Business Capability Modelling

- Leverage the Business Service and Business Service Group constructs to model your organizations capabilities.

- cmdb_ci_service_group parent

- cmdb_ci_service_group children

- cmdb_ci_service_discovered (Service Mapped service)



6. Total Cost of Ownership

- Custom database view set up to allow reporting on all CI's that extend from 'hardware'
- Allows for reports to be run on all assets in a business service that have a cost associated

The screenshot shows the configuration for a database view named 'u_business_services_and_related_computers'. The configuration includes the following fields:

- Name: u_business_services_and_related_computers
- Application: Global
- Label: Business Services and Related Servers
- Plural: Business Services and Related
- Description: (empty text area)

Buttons for 'Update' and 'Delete' are visible below the configuration fields.

Below the configuration is the 'Related Links' section with a 'Try It' link.

The main table displays the view's structure with the following columns: View Tables, New, Go to, Order, Search, and pagination (1 to 3 of 3).

View Tables	New	Go to	Order	Search	1 to 3 of 3
	View = u_business_services_and_related_computers				
		Table	Order	Variable prefix	Where clause
<input type="checkbox"/>		cmdb_ci_service	100	businessservice	
<input type="checkbox"/>		svc_ci_assoc	200	serviceassociation	serviceassociation_service_id=businessse...
<input type="checkbox"/>		cmdb_ci_hardware	300	hardware	serviceassociation_ci_id=hardware_sys_id

At the bottom, there is an 'Actions on selected rows...' dropdown and another pagination control (1 to 3 of 3).

6. Total Cost of Ownership

- Report on all hardware assets within a business service
- Allows for financial data to be leveraged with ServiceNow IT Business Management capabilities

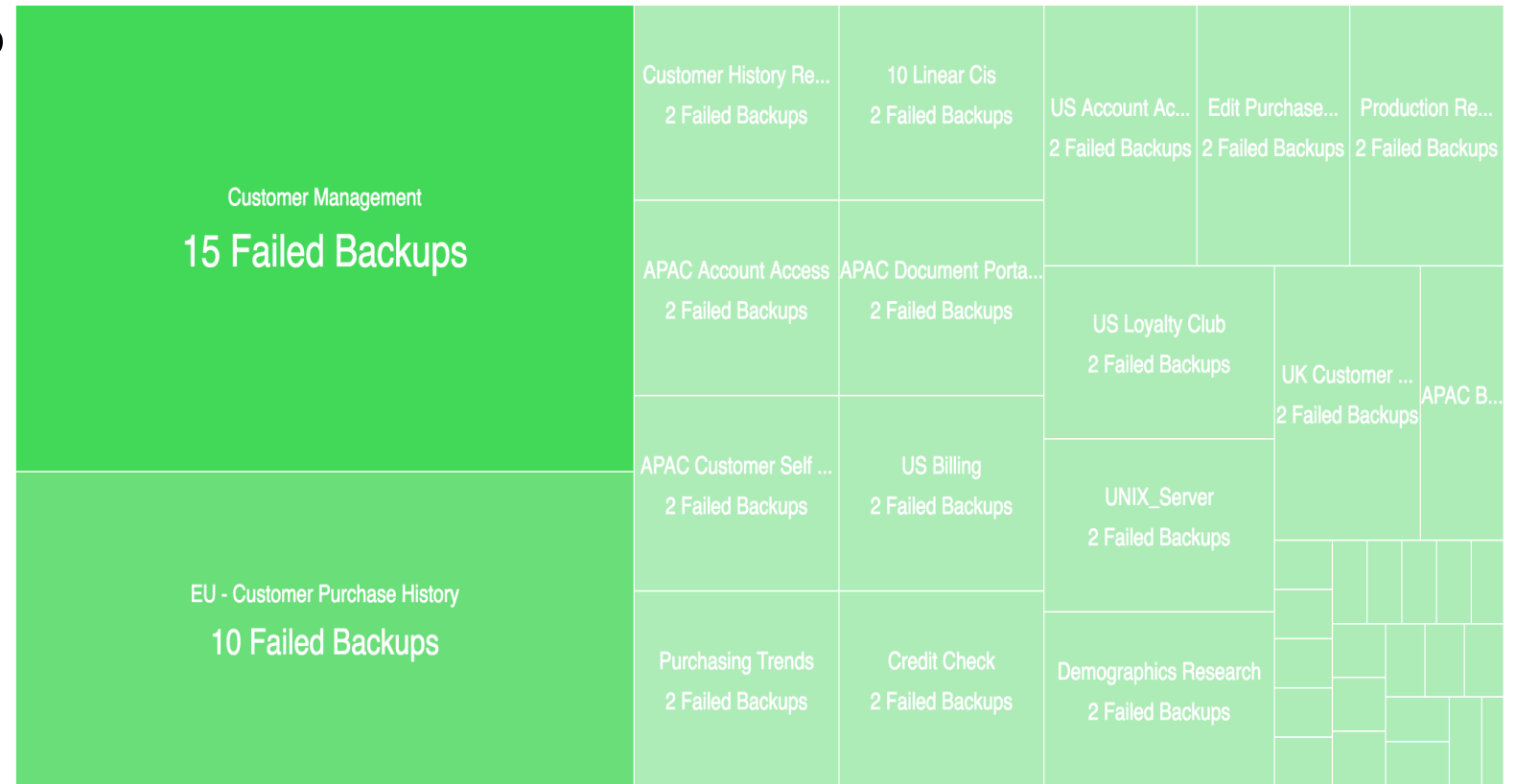
The screenshot shows the ServiceNow report configuration interface for a report titled "Account Access Hardware". The interface includes a header with a back arrow, the report title, an information icon, and "Save" and "Run" buttons. Below the header, there are configuration options for "Data" (set to "Table" and "Business Services and Related Servers [u..."), "Type" (set to "List"), and "Group by" (set to "-- None --"). A "Columns" section is visible, showing a list of available fields such as "Approval group(hardware_change_center)", "Approval group(businessservice_change_center)", "Asset(hardware_asset) [+]", "Asset(businessservice_asset) [+]", "Asset tag(hardware_asset_tag)", "Asset tag(businessservice_asset_tag)", "Assigned(businessservice_assigned)", and "Assigned(hardware_assigned)". A "Selected" section shows the fields currently included in the report: "Name(businessservice_name)", "Name(hardware_name)", and "Class(hardware_sys_class_name)". Below the columns, there are buttons for "Add Filter Condition", "Add 'OR' Clause", and "Add Sort Field". A filter condition is currently set to "Name(businessservice_name) contains Account". At the bottom, a table displays the report results with columns for "Name", "Name", and "Class". The table contains 7 rows of data, each representing an "Account Access" record with a unique ID and a server class.

Name	Name	Class
Account Access	qeretapp22	AIX Server
Account Access	qeretapp21	AIX Server
Account Access	qidbvirt6	Solaris Server
Account Access	qedbvirt3	Solaris Server
Account Access	qeretweb22	AIX Server
Account Access	qedb11	Solaris Server
Account Access	qeretweb21	AIX Server

6. Service Continuity

- By automating the discovery of all the components within a business service, we can start to think about layering different dimensions of data on top
- For example, pulling in all the backup information and storing historical backup state with the CI's.
- We can then report and visualize various business continuity metrics at the business service level.

Continuity - Failed Backups Last Night



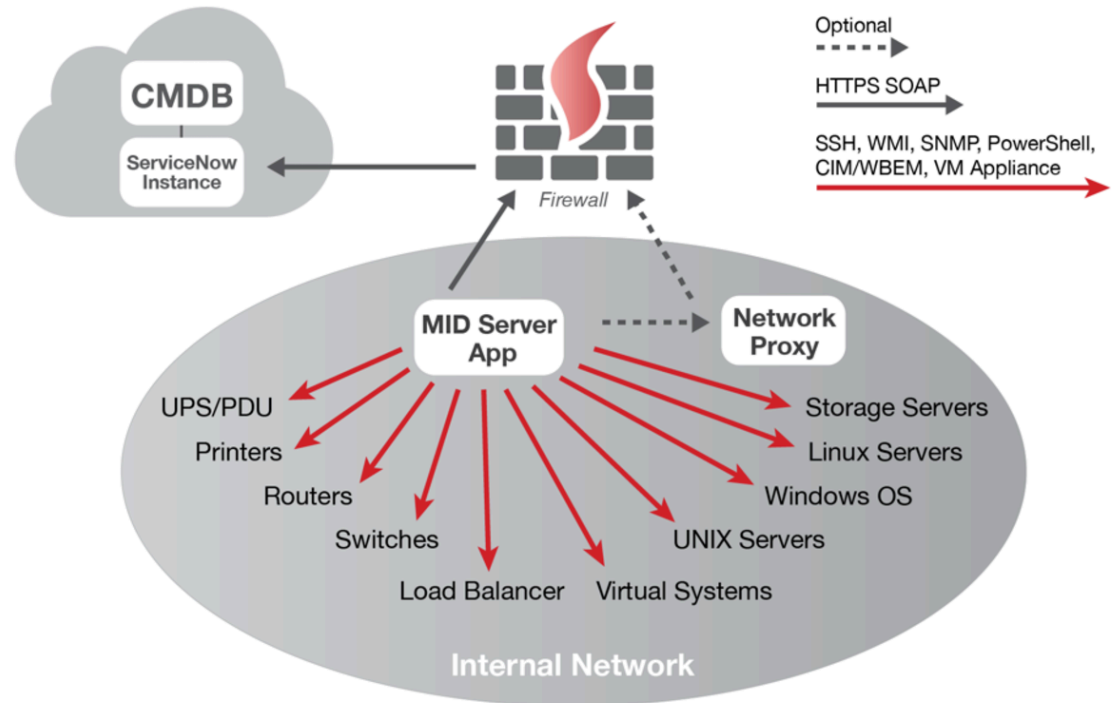
■ 15 - 11 Failed Backups ■ 11 - 8 Failed Backups ■ 8 - 4 Failed Backups ■ 4 - 0 Failed Backups



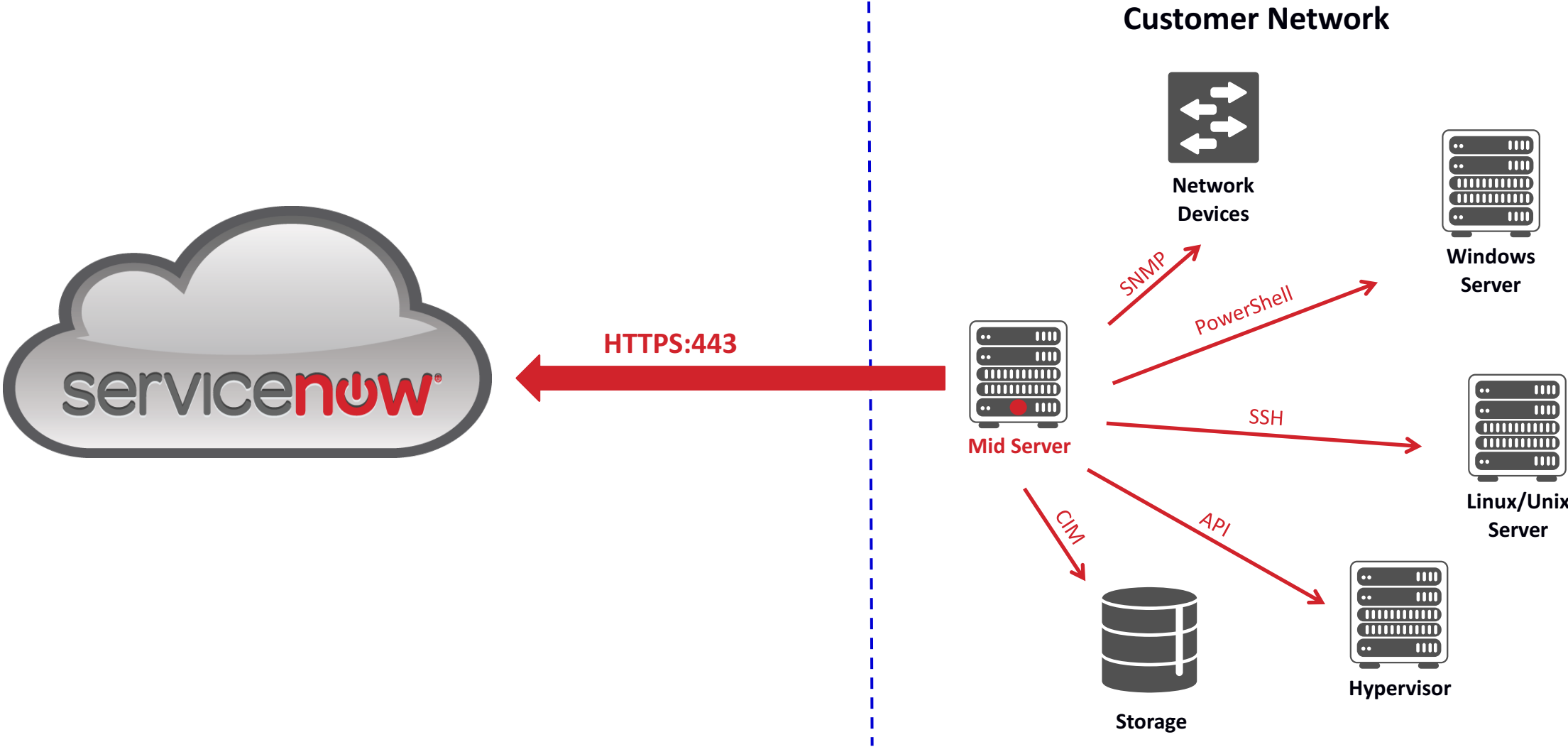
How to plan for a Service Mapping Deployment

Understanding Credentials

- As agentless forms of discovery, both Service Mapping and horizontal discovery require credentials to authenticate against the machines in your environment.
- The types of credentials required will depend on the types of devices you are looking to discover – However there is no magic here, these are normal protocols everyone should understand and be familiar with today e.g. SSH, WinRM, SNMP, etc.
- We do *not* require administrative credentials for most devices. Credentials can be provisioned with the minimum amount of access required to perform the role of discovery.



MID Server Architecture review

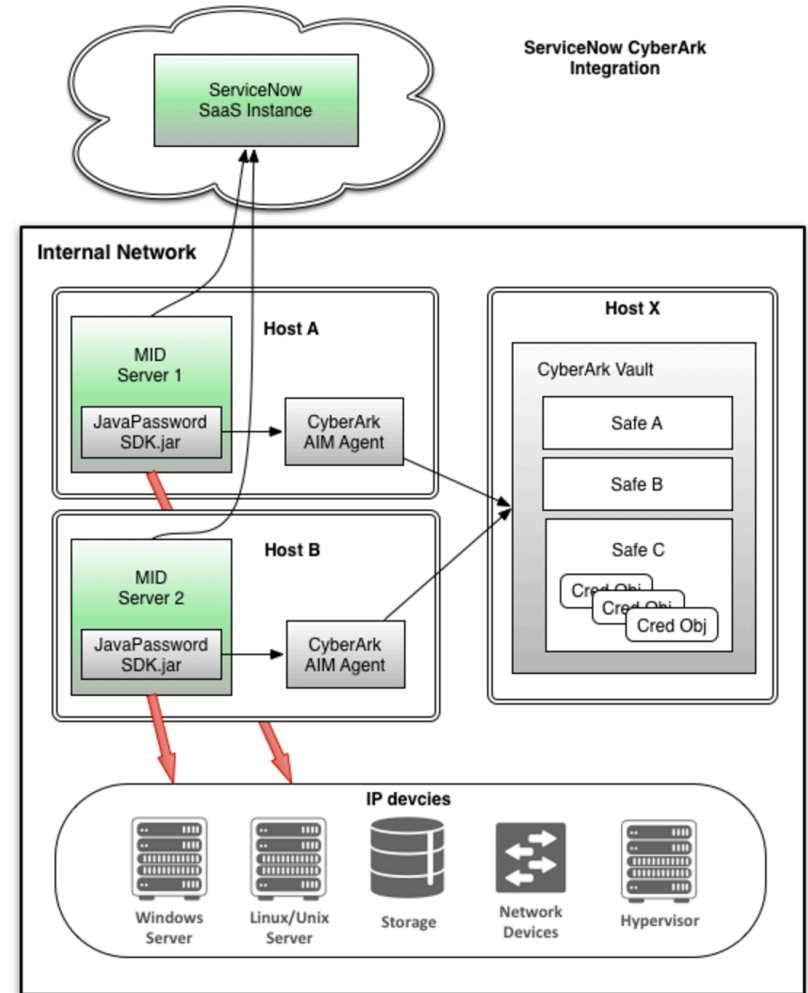


Credential Security

- Only the MID server communicates with target devices
 - No inbound connection required
- MID server communication to server is outbound and always encrypted
 - Credentialed access to individual endpoints -> No agent required
- Credentials are encrypted on the ServiceNow instance
 - Downloaded by the MID server, decrypted in-memory only

Using an External Credential Store

- Some customers with high security requirements simply are unable to put any credentials inside of ServiceNow.
- To help remedy situations like these, we have a framework for using any external, on or off-prem external store a customer may like.
- One such credential store we have out-of-the-box support for is CyberArk.
- Using an external credential store allows the customer to authenticate against, and map their business services without storing any credential information in the ServiceNow instance.



Who Owns the Credentials in my Organization?

- In a large organization it may not be obvious which teams are responsible with helping to fulfil the requirements for a Service Mapping initiative.
- It may be the individual team's responsibility to provision and maintain access. For example the database team may be responsible for access to database servers, the network team to network devices.
- It may be that a centralized account access & security group own the provisioning and maintenance of accounts.



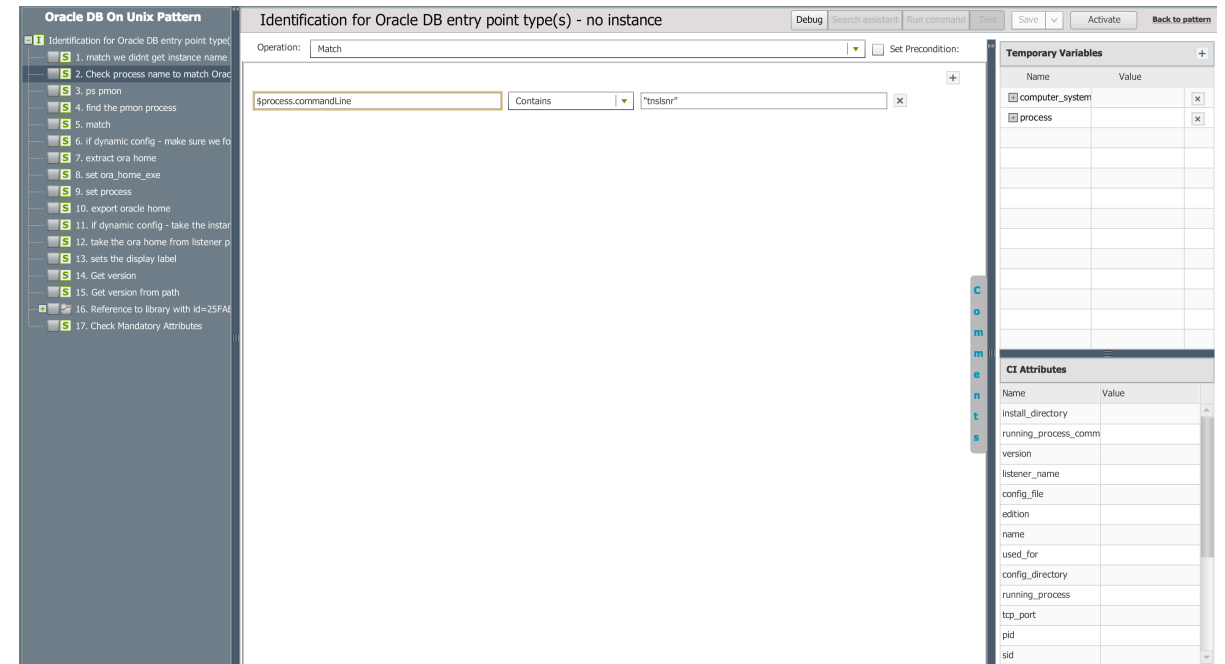
Service Mapping is a Cross-Functional Initiative

- Just like DevOps, a Service Mapping initiative requires members from multiple teams to be successful.
 - Systems Engineering
 - Application Owners
 - Applications Architects
 - CMDB Managers
- These teams all play an important role in the deployment and ongoing maintenance of a service aware CMDB



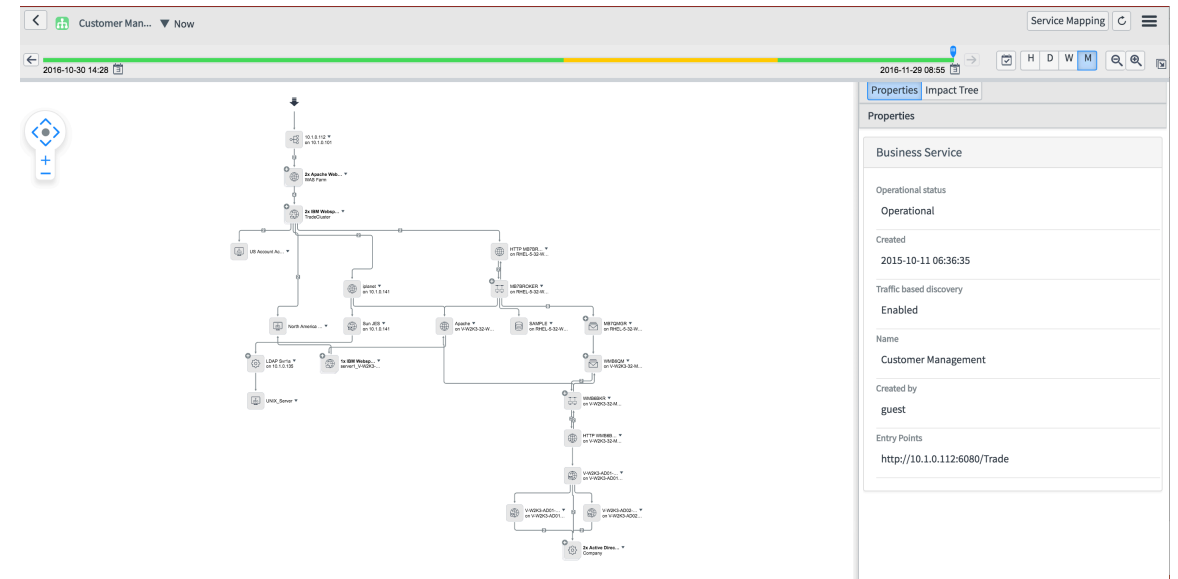
Team Roles – Systems Engineering

- System Engineers or Systems Administrators are often responsible for the hands on configuration or creation of Service Mapping or Discovery patterns.
- While building patterns does not require development expertise, it does require knowledge of Windows and Linux/Unix based systems, basics of networking, and an understanding of IT infrastructure in general.



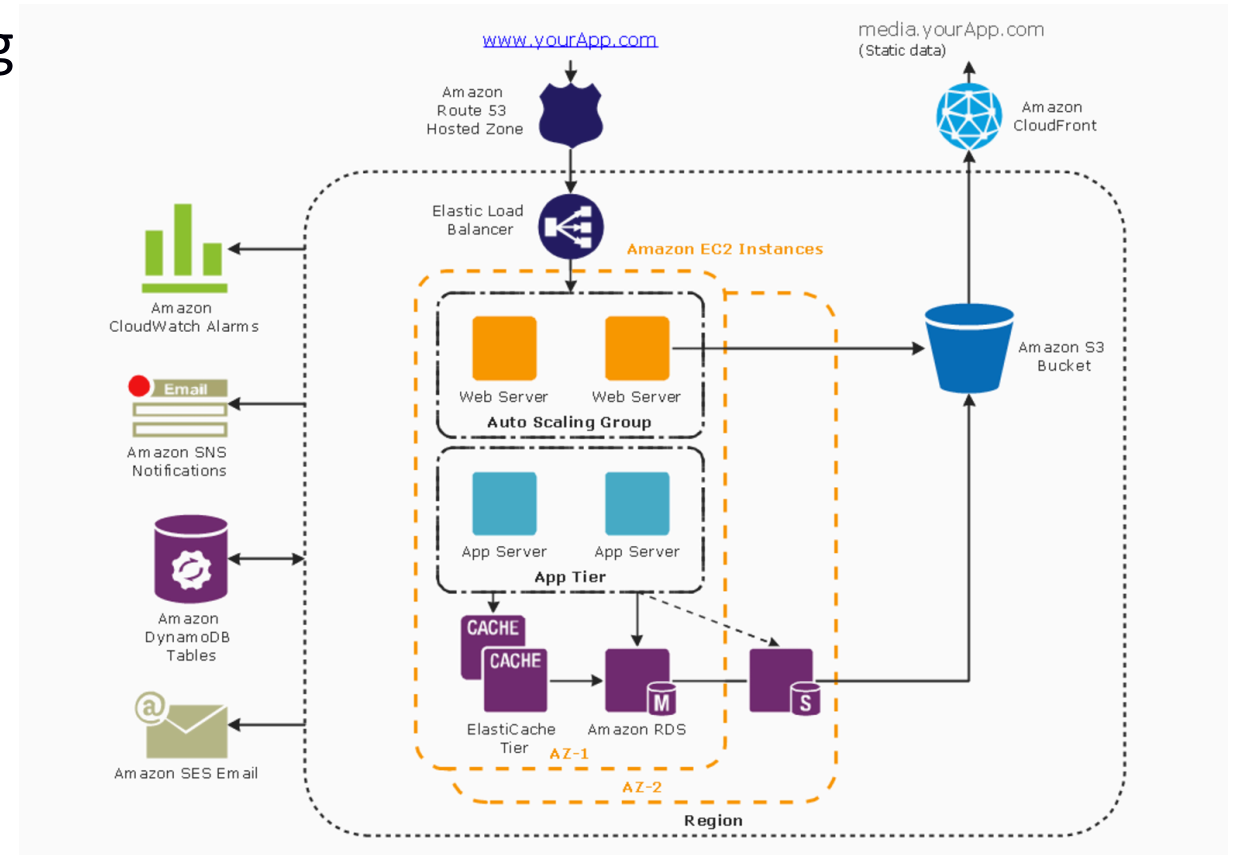
Team Roles – Application Owners

- Application owners are often the consumers of a Service Aware CMDB. You can often think of them like product managers – Ensuring their applications are fulfilling business needs, and delegating to the appropriate engineering teams for new features, or fixing defects.
- It's important they be involved so they can understand the immense benefits they will gain in terms of process improvement from mapping the applications they are responsible for.



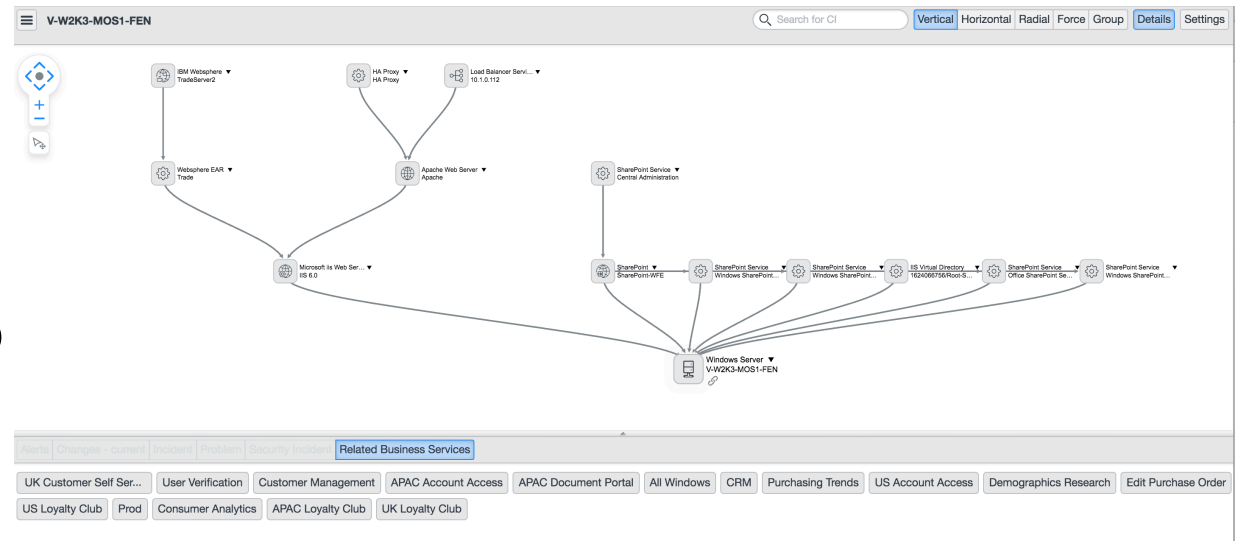
Team Roles – Application Architects

- Application Architects are perhaps the most important role in a Service Mapping initiative. It's the application architects that have designed the systems that are being mapped.
- They understand how the dependencies between components are being defined, what technologies are in use, and any other idiosyncrasies specific to the application architecture.
- They provide input and guidance, as needed, to the Systems Engineers to build and design reusable patterns.



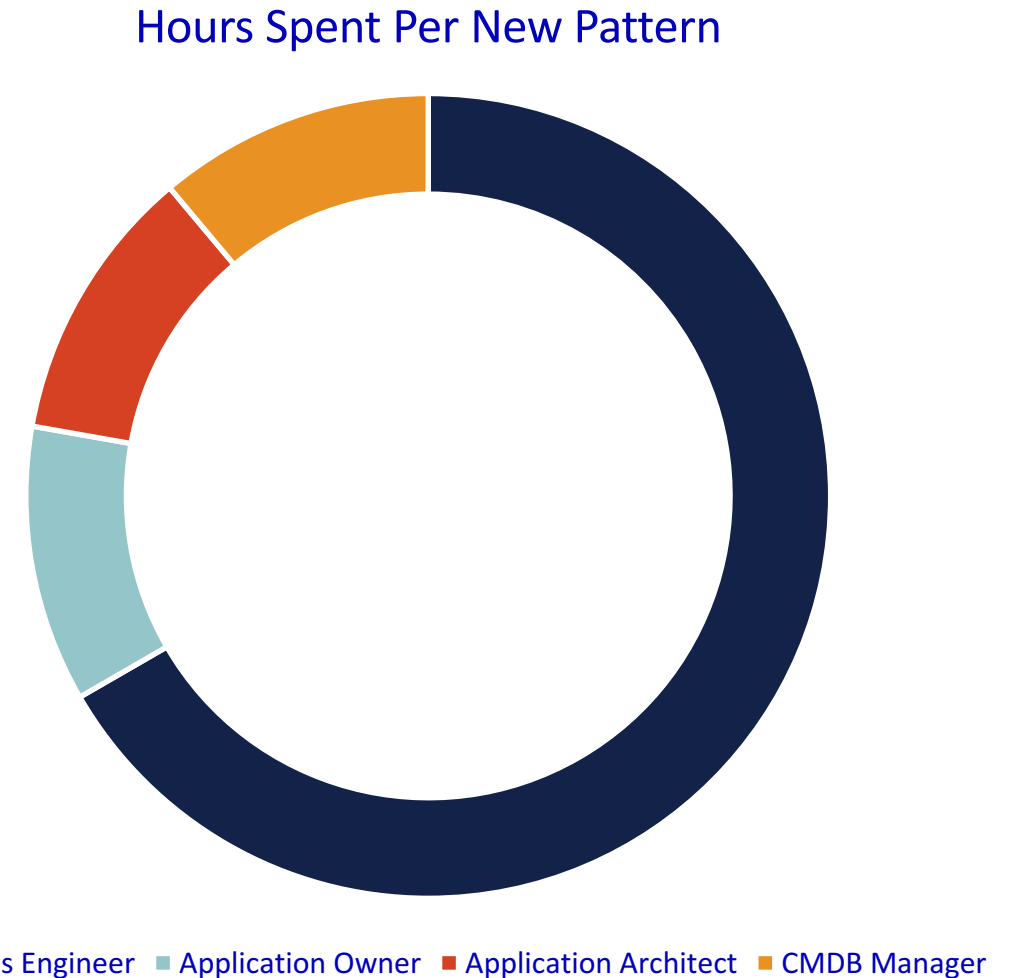
Team Roles – CMDB Manager

- Service Mapping is a CMDB population tool first and foremost. The types of CI's and CI metadata that it stores are varied and complex, allowing for a robust service-aware CMDB.
- It's critical the CMDB Manager understand the changes to the data model that Service Mapping provides, so they can accurately modify any ITSM processes to make more effective use of the CMDB.



Team Roles – Level of Involvement

- You might be thinking, “this sure seems like a lot of different people that need to be involved in the Service Mapping initiative”.
- The reality is each of these team members will spend vastly different amounts of time involved. Some for as little as a few minutes a day, or per pattern/architecture.

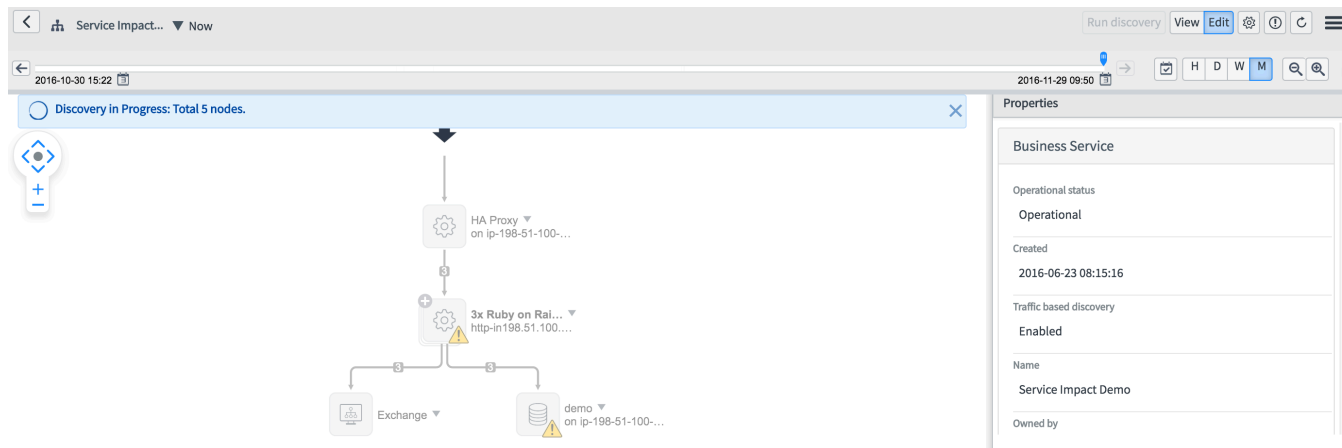




How to Operationalize your Service Mapping Deployment

Keeping Your Service Maps Up-To-Date

- You've gathered the credentials, amassed your cross-functional team and mapped your first business service.
- Congratulations!
- Now how do you operationalize the Service Map? We need to ensure re-discovery of the service runs as often as it needed, that new deployments that may introduce changes to the architecture are accounted for in patterns, and that our IT ops and application owners are getting the maximum value out of the CMDB.



Keeping Your Service Maps Up-To-Date – Discovery Schedules

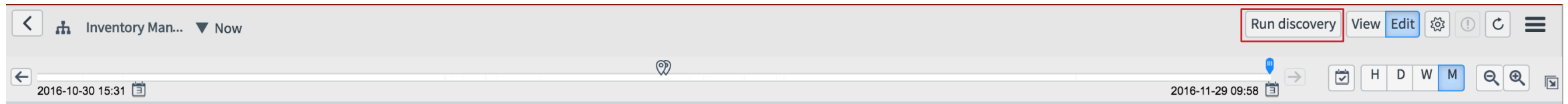
- The default mechanism for re-discovering CI's in ServiceNow is Discovery Schedules
- Discovery schedules allow you to create a user-defined schedule of when re-discovery occurs.
- How often you do re-discovery will likely depend on your organization's use cases, and what your tolerance for staleness is.

The screenshot shows the ServiceNow Discovery Schedules interface. At the top, there is a navigation bar with a menu icon, the text "Discovery Schedules [service_mapping view]", and two buttons: "New" and "Quick Discovery". To the right of these buttons is a "Go to" section with a dropdown menu set to "Name" and a search input field containing the text "Search". Below the navigation bar, there is a filter icon and the text "All > Discover = Service". Below that, there are icons for settings, search, and a dropdown menu set to "Name" with an upward arrow. To the right of this is a "Run" button. Below these elements is a table with two columns: "Name" and "Run". The table has two rows: "All Applications" and "Load Balancer Services", both with a "Daily" frequency. Below the table is a search input field and a dropdown menu labeled "Actions on selected rows...".

	Name	Run
<input type="checkbox"/>	All Applications	Daily
<input type="checkbox"/>	Load Balancer Services	Daily

Keeping Your Service Maps Up-To-Date – On Demand Discovery

- While Discovery Schedules are the default mechanism for keeping Service Maps up to date, there is nothing stopping an operator from running an ad-hoc discovery of a business service at any time (assuming they have the appropriate permissions).
- This may be useful if an IT Operator is about to begin troubleshooting a problem and wants to ensure they have the most accurate, up to date information about the business service.



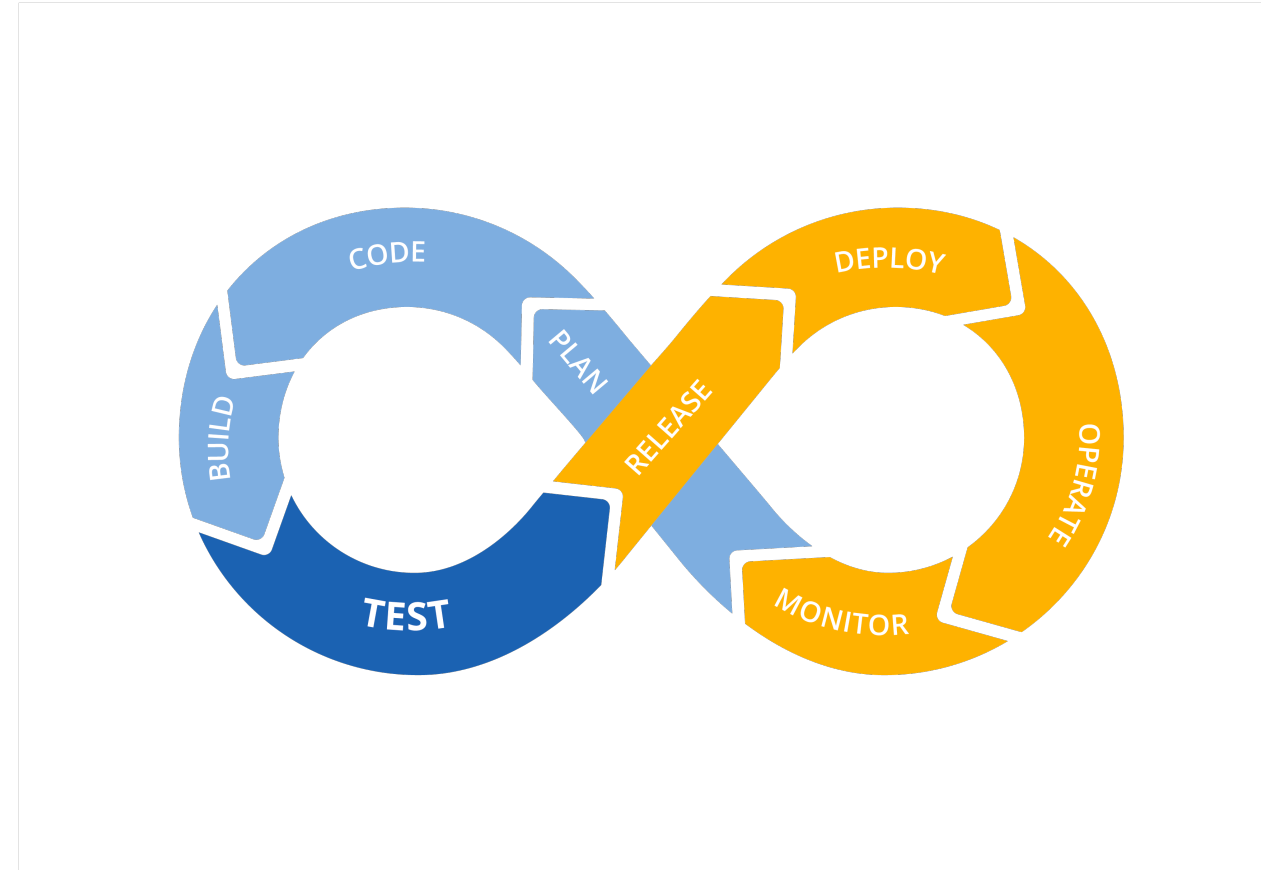
Keeping Your Service Maps Up-To-Date – APIs and Automation

- For some organizations schedules and ad-hoc discovery are not enough to ensure that Service Maps are up to date.
- Perhaps you are developing a brand new CloudNative application on AWS and are leveraging things like auto-scaling.
- You can leverage the ServiceNow APIs to programmatically discover or re-discover a business service, should you have a use case that requires that level of automation.

```
(function executeRule(current, previous /*null when async*/) {  
  
    var serviceId = getDiscoveredServiceId();  
    if (gs.nil(serviceId)) {  
        return;  
    }  
  
    var bsManager = new SNC.BusinessServiceManager();  
    // clear the business service  
    bsManager.clear(serviceId, false);  
    // start the discovery  
    bsManager.startDiscovery(serviceId);  
  
    function getDiscoveredServiceId() {  
        var serviceGr = new GlideRecord('cldb_ci_service_discovered');  
        serviceGr.addQuery('name', current.name);  
        serviceGr.query();  
        if (serviceGr.next()) {  
            return serviceGr.sys_id;  
        }  
    }  
  
})(current, previous);
```

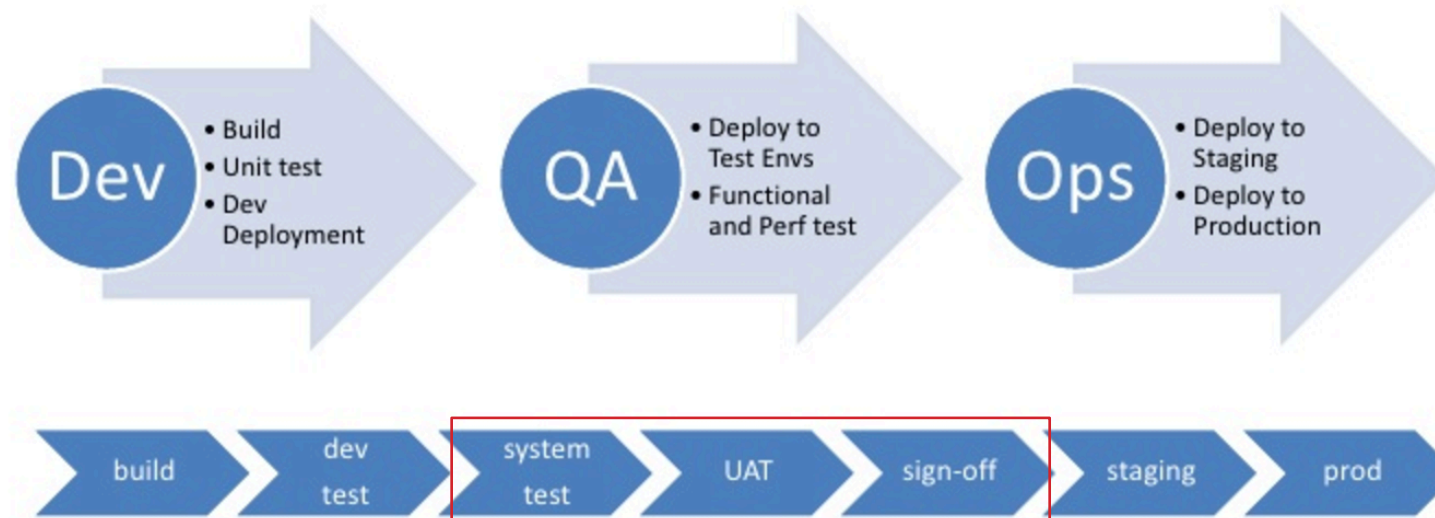
Making Service Mapping Part of your SDLC

- Keeping Service Maps up to date is only ½ the battle, we also need to ensure that newly provisioned services and applications, as well as major architectural changes to existing services are captured.
- These types of changes *may* require new Service Mapping patterns to be developed, or existing patterns to be modified.



Making Service Mapping Part of your SDLC

- If we visualize a typical enterprise Software Development Lifecycle it becomes easier to see where we may want to put stopgaps to ensure that new Service Mapping patterns are developed, or existing ones modified
- Where your organization decides to perform that action is up to you, but we think that the QA function ties in very closely to Service Mappings goals and values.



Service Mapping Attestation

- Being a prescriptive approach to application dependency mapping – Meaning that we are writing and modifying patterns to tell Service Mapping how to discover our architectures, it's important to ensure that the maps accurately depict said architectures.
- This is a critical piece to completing a service mapping initiative, whether it's the first map in your organization, or the 50th.
- This is something your team's application architects can help you with, as they are the ones most intimately familiar with it.



now
Thank you



servicenow®

**Taking Your Service Management
To The Next Level!**

Expertise

Formerly known as Configure Consulting, ConfigureTek is a full-service technology provider empowering IT Operations organizations with the efficiency, stability, and top performance in the delivery of business services in just weeks instead of years.



CMDB



Service Mapping



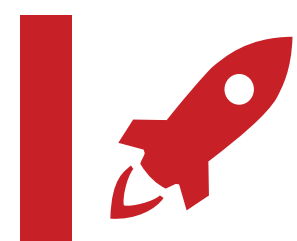
Asset Mgmt



Software License
Compliance

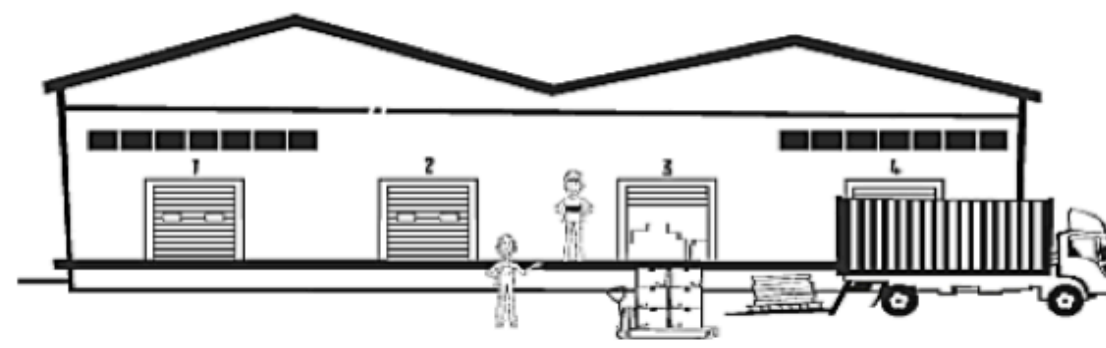
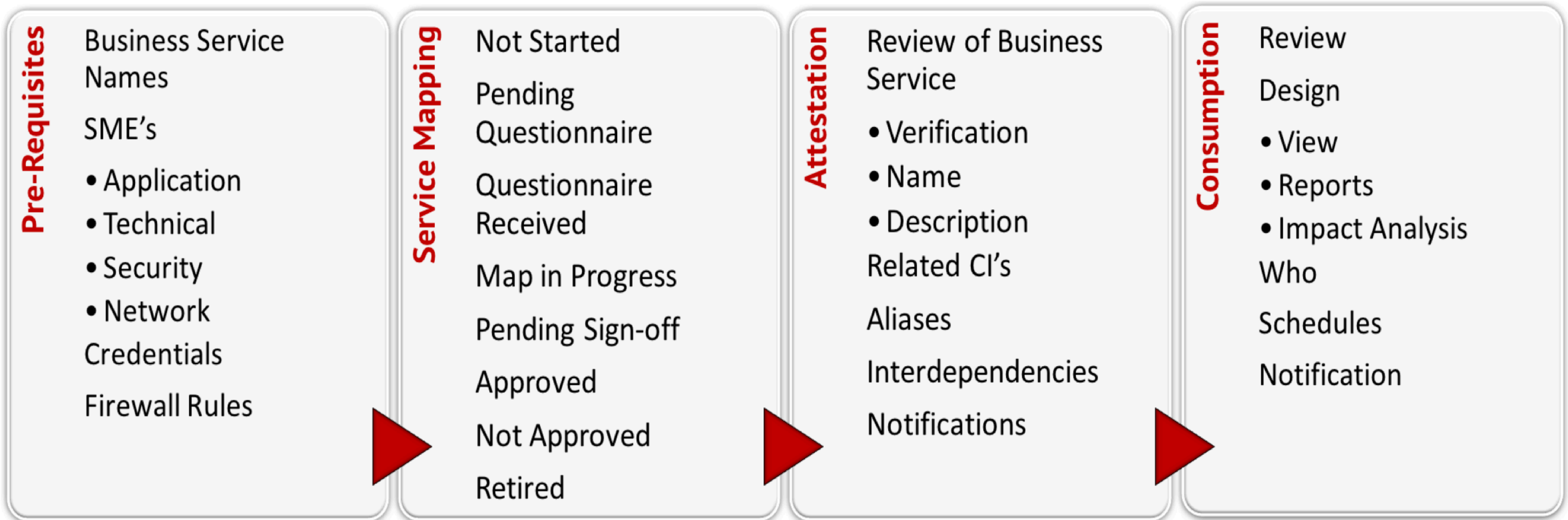


Event Mgmt



Quick-Start Solutions: Rapid Service Mapping Factory, Rapid Discovery, CMDB, Asset, MOM

Service Mapping Factory - Methodology



Service Mapping Accelerators

Lifecycle

Service Mapping Lifecycle Tracking. A proven and repeatable methodology for tracking your organization's service mapping lifecycle.

Name	View map	Service Mapping Lifecycle	Business criticality	Owned by
!Colin	Search	Search	Search	Search
BSM	View map	Not Started	4 - not critical	Abel Tuter
CM	View map	Pending Sign-off	4 - not critical	Erol Luddy
CMDB	View map	Pending Questionnaire	1 - most critical	Tim Robinson
EmployeeServices	View map	Mapping In Progress	3 - less critical	Erol Luddy
OOProd	View map	Retired	3 - less critical	Timothy Janski
Payments	View map	Pending Questionnaire	1 - most critical	Timothy Janski
Payroll	View map	Mapping In Progress	2 - somewhat critical	Tim Robinson
RPCClient	View map	Pending Questionnaire	4 - not critical	Timothy Janski
Service Health Reporter	View map	Pending Questionnaire	2 - somewhat critical	Abel Tuter

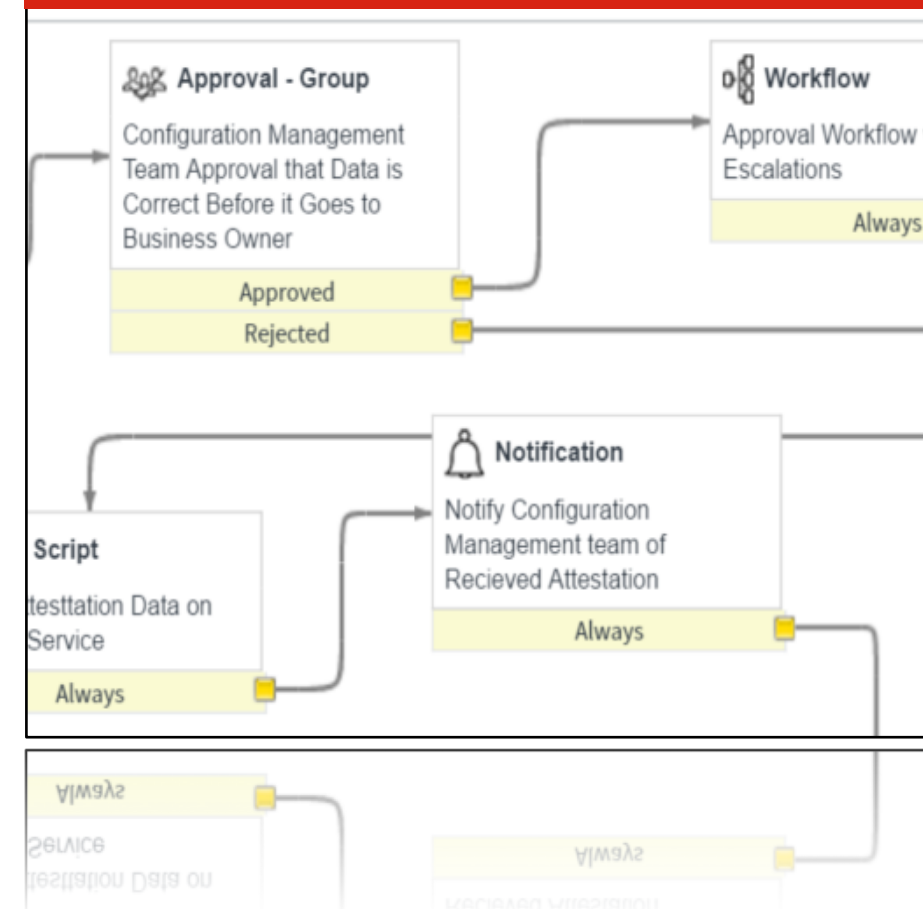
Questionnaire

Automated SME questionnaire distribution. Pre-created in-tool questionnaires to enable your service and app mapping project.

Number: 1000000000
Assigned to: Tim Robinson
Assignment group: Configuration Management Team
Business service: CMDB
Questions: CMDB Management Database
Additional People: Ask Philip
How to the business service address (URL, email, phone, or IP address)? Place
Add the email users external or internal?

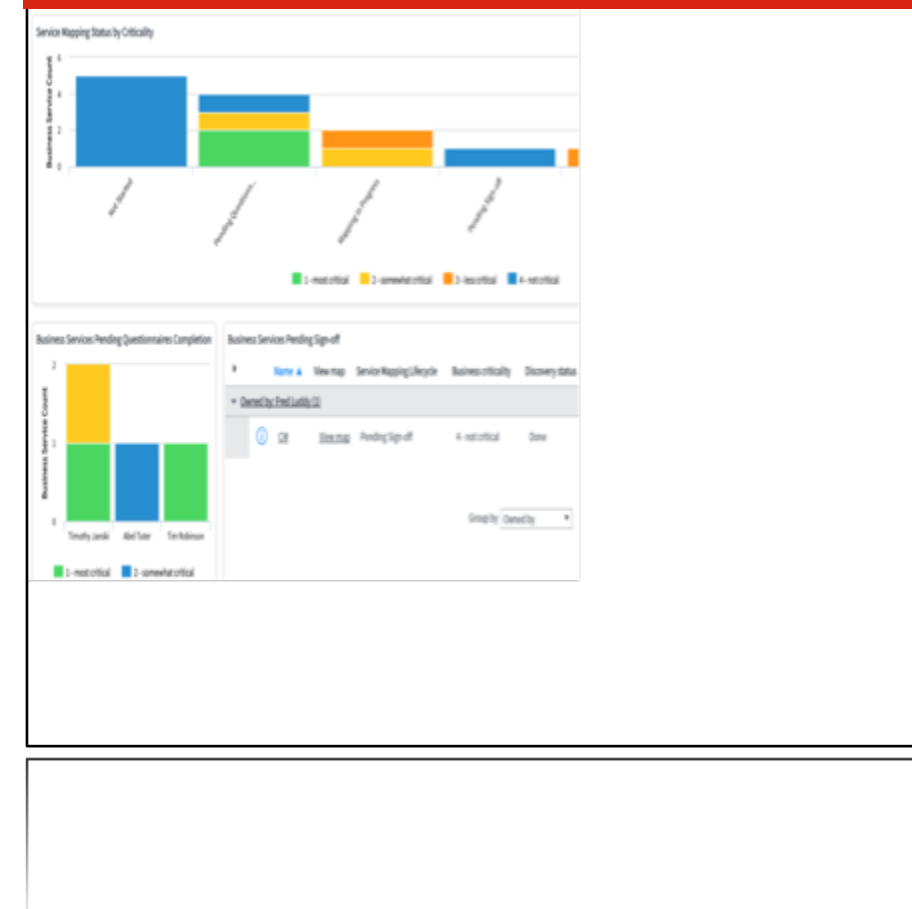
Attestation

Pre-configured SME validation and verification workflow for service mapping projects.



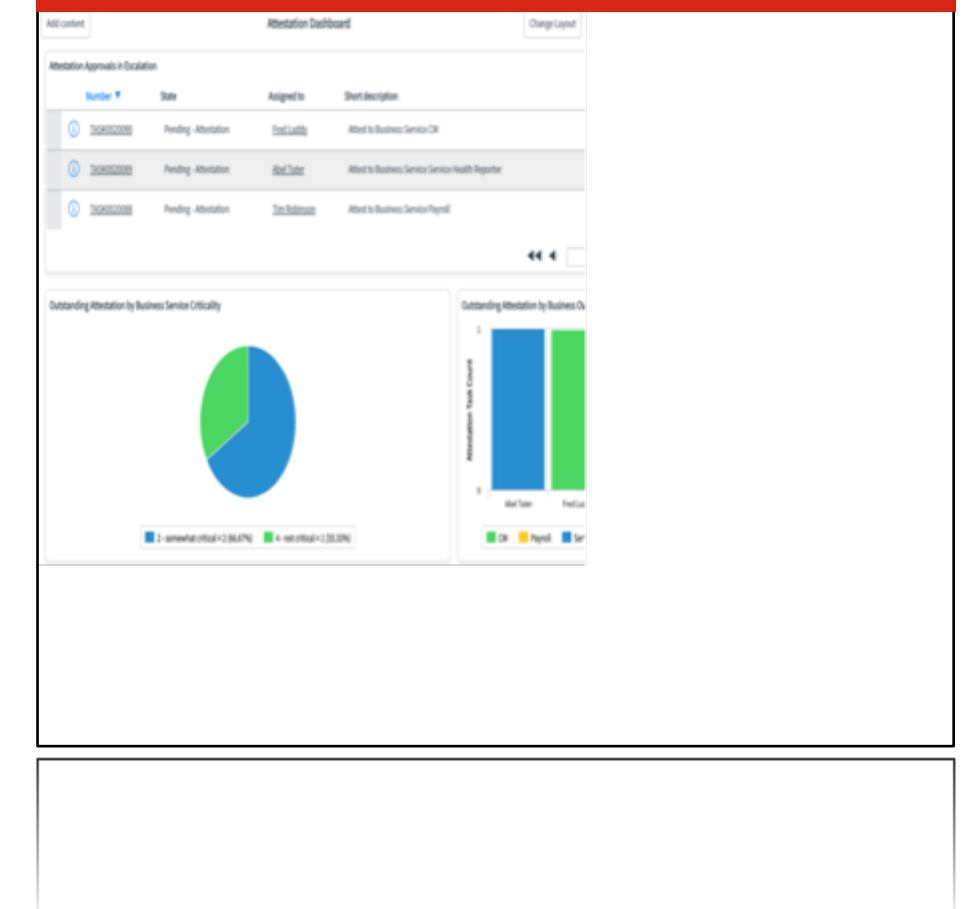
Dashboard

Track and visualize your organization's service mapping progress.



Reporting

Track and visualize completed and pending attestation to accelerate service mapping.



Implementation Accelerators



Pre-Configured Collateral

ConfigureTek enables turnkey implementations and instant ROI by providing pre-configured accelerator collateral.

AWARD WINNER
Solution Growth Partner



- Pre-engagement check list for infrastructure, access, security
- Pre-engagement guidelines on architecture, sizing, security
- Pre-engagement credential requirements guidelines
- Auto discovery best practices guidelines
- Pre-configured auto discovery packaged modules
- Service Mapping tracker for mass mapping projects
- Service Mapping pre-configured questionnaires for SME input
- Sample and template project plans for quick enablement
- Training courses designed for customers to achieve quick ROI
- Pre-configured templates for post-engagement documentation

Service Mapping Lifecycle & Progress Tracking



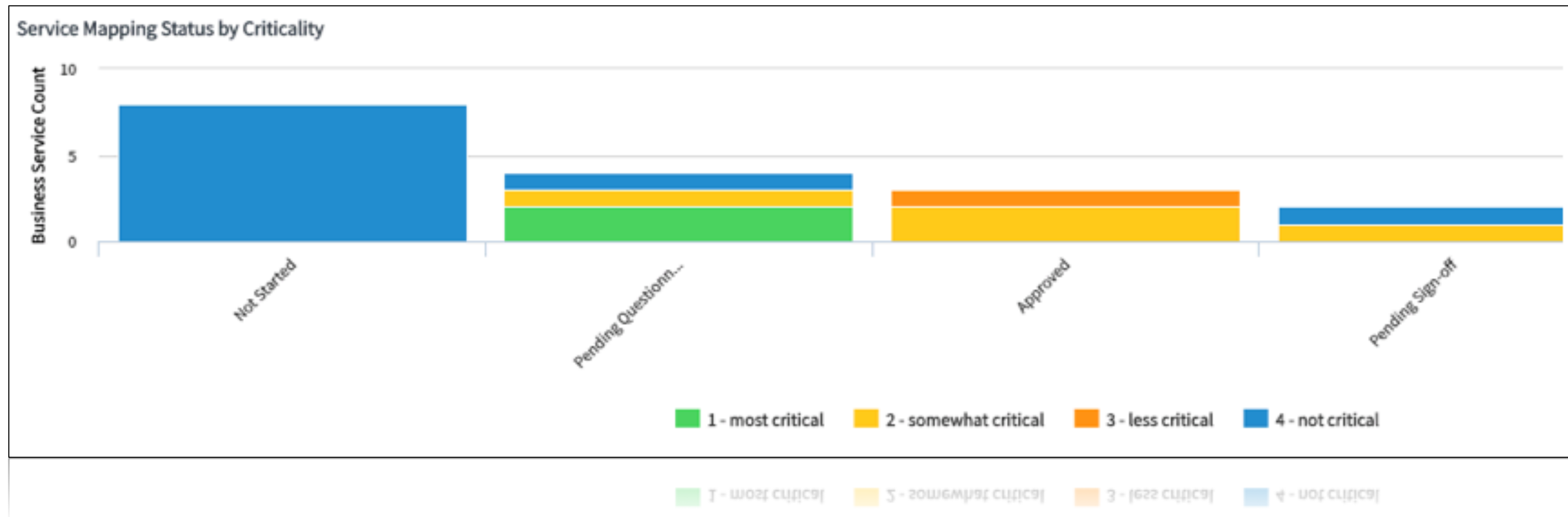
Workflow Automation
Automatic tasks automation to progress through the lifecycle.



Bottleneck Resolution
Automated bottleneck resolution and escalation algorithm.



Task Assignment
Automatic task assignment to SMEs and App Owners



Service Mapping Approvals, Bottlenecks, & Escalations

Attestation Pending Approval More Than 5 Days Old

Number	Priority	State	Assigned to	Short description	Task type	Workflow activity
TASK0020090	4 - Low	Pending - Attestation	Fred Luddy	Attest to Business Service CM	Attestation Task	
TASK0020089	4 - Low	Pending - Attestation	Abel Tuter	Attest to Business Service Service Health Reporter	Attestation Task	

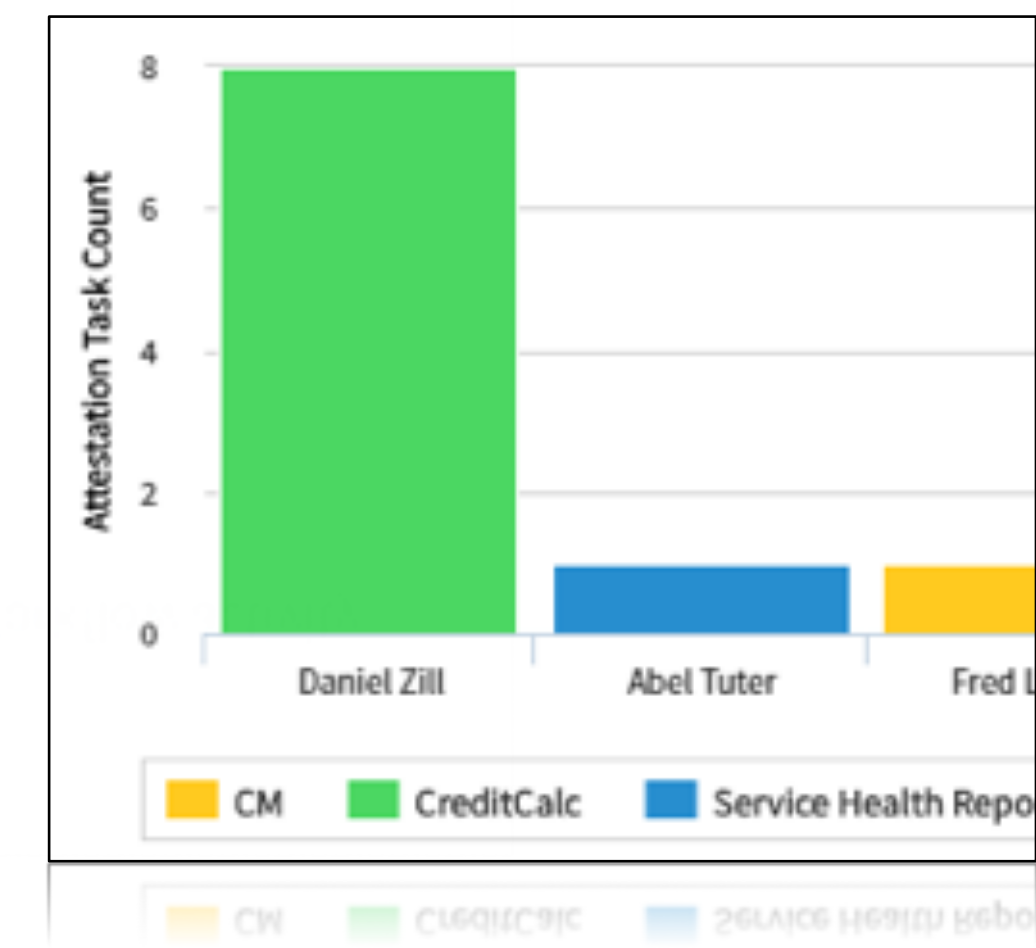
1 to 2 of 2

Attestation Approvals in Escalation

Number	State	Assigned to	Short description	Task type
TASK0020090	Pending - Attestation	Fred Luddy	Attest to Business Service CM	Attestation Task
TASK0020089	Pending - Attestation	Abel Tuter	Attest to Business Service Service Health Reporter	Attestation Task

1 to 2 of 2

Approval Queue by Biz Owner

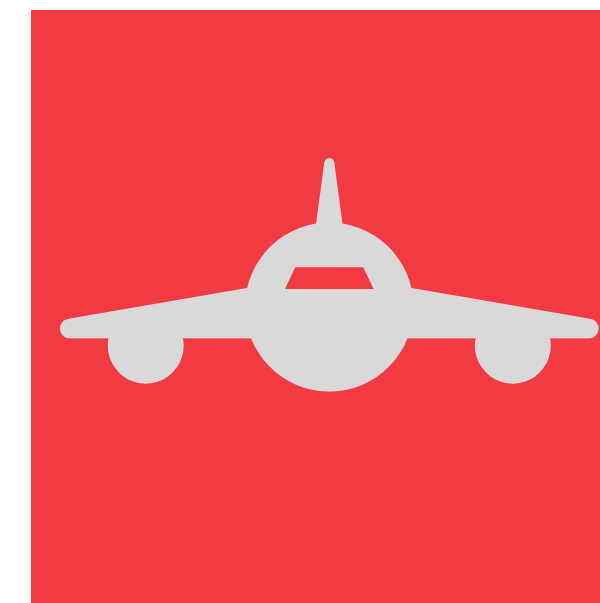


Why ConfigureTek



Sales

RFPs, Sales, Demos, POVs, POCs,
Business cases, Proposals



Implementations

SOWs, Best practices, Accelerators,
Project plans, Architecture, Security



Administration & Staffing

Administration of ITOM, Staff
Augmentation



Knowledge & Expertise

Strong team of engineers,
consultants, sales, and marketing