**INTERMOUNTAIN HEALTHCARE**

**Platform X – Dynatrace to IM Service Hub Integration**

**CONCEPTUAL TECHNOLOGY MODEL**

**Author: James (Jim) Hall**

Version 0.1

25 April 2024

Important Notice

This document has been prepared by DXC. The contents are confidential and must not be communicated in whole or in part to any other party without the prior written approval of DXC.

No copy or other reproduction, other than for the purpose of evaluation of this document, shall be made in whole or in part without the prior written permission of DXC.

DXC has prepared this document in good faith based on the information made available to it. Many factors outside DXC's current knowledge or control may affect the recipient's needs and plans. The statements in this document are qualified accordingly. Nothing in this document or in any related discussions or correspondence shall be construed as an offer, or the basis of any contract, nor may a representation which may be relied upon by any person except as DXC expressly agree in writing.

Document Purpose

The Conceptual Technology Model outlines the conceptual architecture solution that meets the known requirements of the client. This includes:

* A statement of the client requirements.
* Conceptual details of all the logical devices, and logical locations, mapped to specific functionality, rather specific technical infrastructure.
* An overview of how the solution fits together at a conceptual level.

Table of Contents

1. Introduction 6

2. Scope and Requirements 7

2.1. Scope 7

2.2 DXC Detect to Correct 8

2.3 Incident Ebond 9

2.4 CONFIG API FOR DECOM, AGENTLESS, AND SYNTHETIC PING UPDATES TO DXC SERVICE NOW 10

2.5 Maintenance Window API 13

3. Conceptual Architecture Environment 14

3.1. Current Conceptual Design 14

3.2. Target Conceptual Design 15

4. DXC EDGE TO DXC PLATFORM X DATA CLASSIFICATION 16

5. Assumptions, Risks and Dependencies 17

5.1. Assumptions 17

5.2. Risk Register 17

5.3. Dependencies 17

Appendix A Control 18

Document Authorization 18

Document Distribution 18

Change History 18

Document References 18

Exception Request Approvals 19

Appendix B Glossary of Terms 20

Table of Figures

[Figure 1: Process Flow Diagram of Solution 7](#_Toc110332807)

[Figure 2: Detect to correct 8](#_Toc110332807)

[Figure 3: CuRrent State Design 14](#_Toc110332808)

[Figure 4: Future state design 15](#_Toc110332809)

Table of Tables

[Table 1: Data Classification 16](#_Toc110332813)

# Introduction

Today Intermountain Healthcare uses Zabbix for monitoring the DXC managed servers. Keeping Intermountain Healthcare (IH) systems updated with the latest monitoring is essential for smooth and secure operation of the systems. The goal of this solution is to ensure alignment with Intermountain DTS strategies to simplify, modernize, and transform the tools.​ Platform X - Edge X 2.0 will provide a standardization framework by leveraging current investments (capabilities and processes) which gives us a head-start in implementation.

DXC Platform X, next generation of DXC AIOPS solution (data centric, intelligent automation platform). This will significantly simplified the enterprise platform from platform x with use of it best in enterprise-class products from Dynatrace, ServiceNow, Raffia, and DXC.

# Scope and Requirements

## Scope

The goal of this project is to deploy the Platform X – Edge X 2.0 Tools (Dynatrace and EKAM) and deploy Dynatrace OneAgents on all DXC managed servers with the exception of any servers that fall under the separate "Grail" reporting activity managed by Mike Woods and **add Oracle DB monitoring** , DXC will begin use of the Dynatrace/EKAM application and agents for monitoring and maintenance across supported DXC managed servers.

IH ServiceNow instance is the source of truth. DXC delivery teams will work out of the IH ServiceNow instance. Whenever a change window is created in the IH ServiceNow instance, an API call will be made to DXC Dynatrace to insert a mantenance record (approved change window) for the particular CI, which would control event and automation suppression within Dynatrace with exception of long window approved changes (Patching, Decommission, Project related changes) which will follow the manual suppression process by delivery teams. All incidents that require manual remediation by the delivery team member will be sent to IH ServiceNow through the incident ebond. Discovery will create and update CI records in both IH ServiceNow and DXC ServiceNow. DXC will use the config API between Dynatrace and DXC ServiceNow to modify the CI records for Decom, Agentless, and Ping. DXC will setup an automated provision/deprovision scripting to modify DXC ServiceNow and utilize the NEXO enrolment. Config API will run from the EKAM server. DXC will run SGO job to automatically create CI in service now for Agentless. DXC will use lDSS load sheets to create the CI for ping and use them to update agentless and make updates to retire CI's. Below represents the process flow diagram for the integration between Dynatrace/DXC ServiceNow and IH ServiceNow.

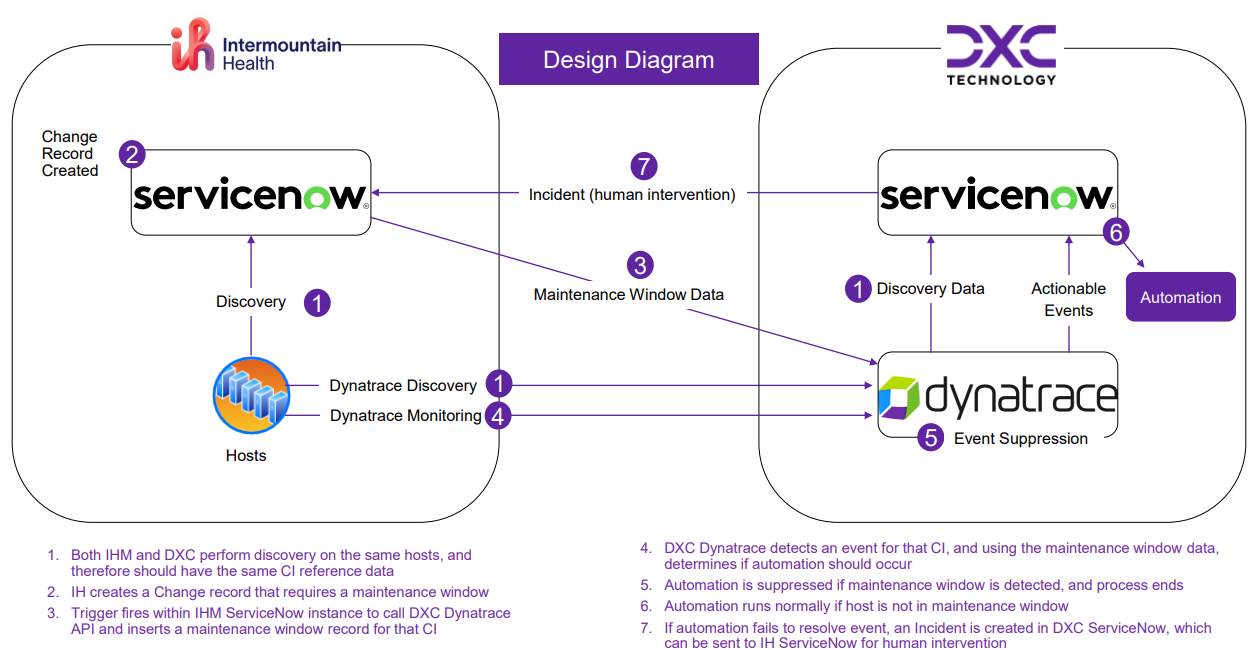


Figure 1: Process Flow Diagram of Solution

Process Flow:

1. Both IH and DXC perform discovery on the same hosts, and therefore should have the same CI reference data
2. IH creates a Change record that requires a maintenance window
3. Trigger fires within IH ServiceNow instance to call DXC Dynatrace API and inserts a maintenance window record for that CI
4. DXC Dynatrace detects an event for that CI, and using the maintenance window data, determines if automation should occur
5. Automation is suppressed if maintenance window is detected, and process ends
6. Automation runs normally if host is not in maintenance window
7. If automation fails to resolve event, an Incident is created in DXC ServiceNow, which can be sent to IH ServiceNow for human intervention

## DXC DETECT TO CORRECT

[**ONBOARDING INSTRUCTIONS**: Update the following host table for all Activegates deployed only. Each Activegate is identical, therefore no need to repeat the configuration sections of the table below]

Using Detect-to-Correct automated tasks to undertake corrective actions such as freeing up disk space helps to reduce cost and human input.

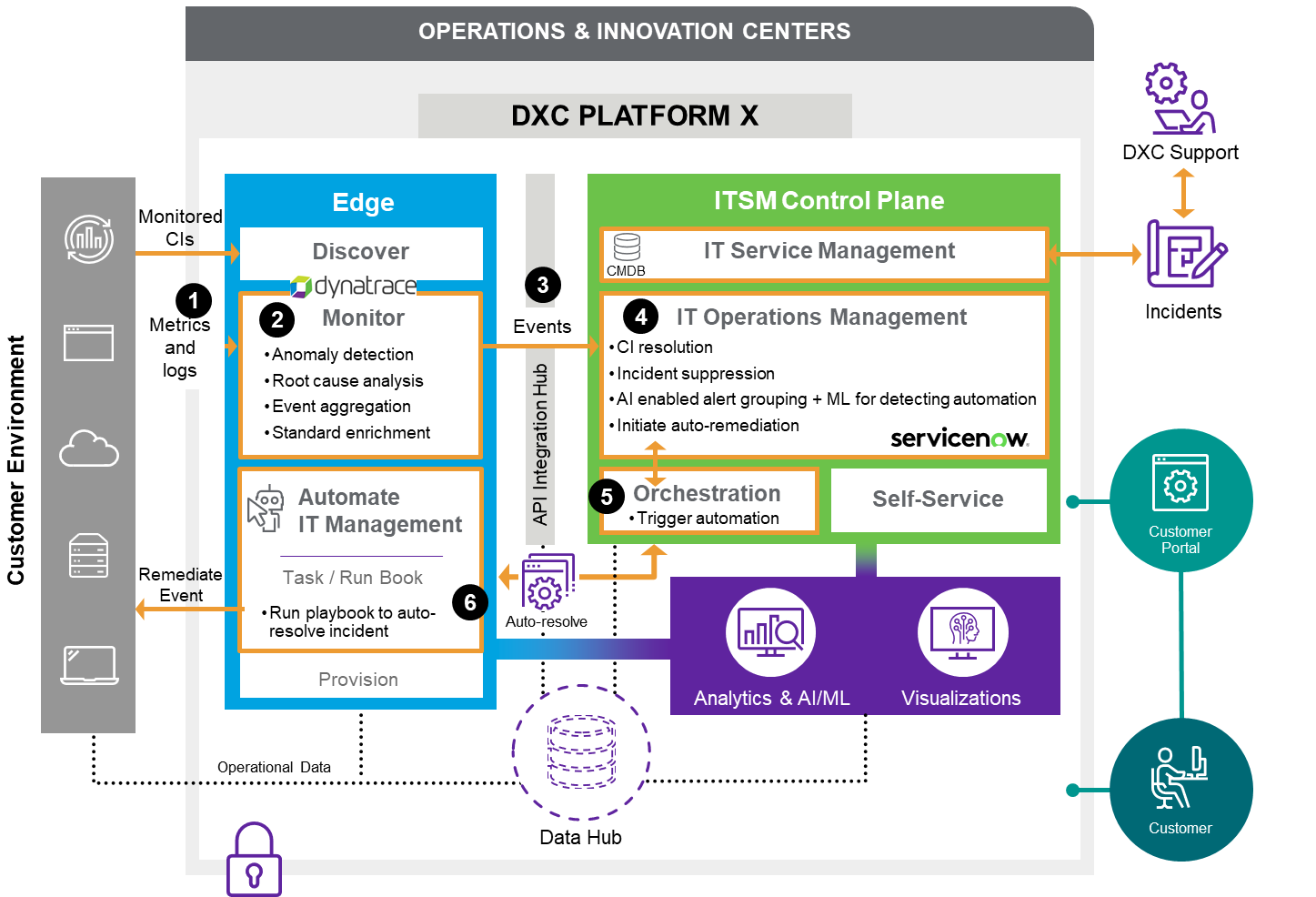


Figure 2: Detect To Correct

Primary function is to facilitate onboardings and adapt to customer environments. This service provides a lightweight, flexible and adaptive automation solution that can be implemented quickly and utilized with minimal instruction.

Primary Objective:

1. Metrics and Logs are collected from the customer environment into the Dynatrace Monitoring Environment
2. The Monitoring Environment AI engine continually assesses the data for significant events.
3. Only significant events are sent to ServiceNow IT Operations Management
4. DXC ServiceNow IT Operations Management capabilities assess the significant event alongside other inputs
5. Through the workflow capability of Orchestration, an automated solution to the significant event is triggered
6. This results in an automated resolution to the event and associated incident

## Incident Ebond

**DXC to IH Assignment Group Mappings:**

|  |  |
| --- | --- |
| **Platform X ServiceNow Group** | **IH ServiceNow Group** |
| iSolve Database ICI Oracle | Can - DBA Oracle |
| iSolve Middleware ICI | Can - Middleware |
| iSolve UNIX ICI | Can - Unix System Admin (SA) Operations |
| iSolve Storage and Backup ICI Storage | Can - Storage System Admin (SA) |
| iSolve Windows ICI | Can - Windows System Admin (SA) Operations |
| iSolve Database ICI MSSQL | Can - DBA SQL |
| iSolve UNIX ICI AS400 | Can - AS400 Support |
| iSolve Storage and Backup ICI Backup | Can - Backup System Admin (SA) |
| [IM OpenShift](https://nam12.safelinks.protection.outlook.com/?url=https%3A%2F%2Fdxcdomprdus.service-now.com%2Fsys_user_group.do%3Fsys_id%3Ddc6844c1879fb09c9a6f42ed0ebb359b%26sysparm_record_target%3Dsys_user_group%26sysparm_record_row%3D1%26sysparm_record_rows%3D1%26sysparm_record_list%3DnameCONTAINSOpen%255EORDERBYname&data=05%7C02%7Cjames.hall%40dxc.com%7C39ea7416f08345a7c3b808dc54bd9318%7C93f33571550f43cfb09fcd331338d086%7C0%7C0%7C638478420403716882%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C0%7C%7C%7C&sdata=8kRLBMqXz%2Flm8yqzaNPO%2B4pPtZ%2BNgoHgdTcVPln0RVk%3D&reserved=0) | Can - Openshift |
| BigFix Support ICI | Can - DXC IM BigFix Support |
| Service Desk | Can - Service Desk |

**Requirements List:**

* CIs should be passed from DXC to IH without fail, for all Active ebonded incidents
* Incidents -
  + Incidents auto-resolved by the client approved automation playbooks within DXC ServiceNow will not be ebonded to IH ServiceNow
  + Any incident not auto-resolved by the client approved automation playbook will be regarded as needing manual remediation and ticket will be ebonded to IH ServiceNow.
  + Incidents needing manual remediation by DXC delivery teams will be ebonded to IH ServiceNow
  + Any incident not auto-resolved by the automation playbook will be regarded as needing manual remediation. When the issue resolves, either by itself or by the DXC delivery team, the Dynatrace alert will close, thus resolving the DXC ServiceNow incident
  + The incident state upon creation needs to be passed from DXC to IH as NEW.
* Work notes / Additional comments should be ebonded - bi-directional
* Change in ticket states should be ebonded - bi-directional (Network)
* Priority changes should be ebonded - bi-directional
* Required fields for Incident:
  + Short Description
  + Description
  + Caller
  + Category
  + State
  + Urgency
  + Priority
  + Configuration Item
  + Assignment Group

## CONFIG API FOR DECOM, AGENTLESS, AND SYNTHETIC PING UPDATES TO DXC SERVICE NOW

To integrate Dynatrace configuration API with DXC ServiceNow for decommissioning, agentless monitoring, and synthetic ping updates, you would typically follow these general steps:

Authentication Setup:

Obtain the necessary credentials (API tokens, usernames/passwords, or other authentication mechanisms) from both Dynatrace and DXC ServiceNow to authenticate your requests.

Decommissioning Configuration:

Utilize the Dynatrace configuration API to remove any monitoring configurations for decommissioned services, hosts, or applications.

Agentless Monitoring Setup:

Configure Dynatrace to monitor services without requiring an agent installation. This might involve setting up synthetic monitors or utilizing OneAgent deployment modes like "OneAgent for web server" or "OneAgent for ActiveGate".

Synthetic Ping Updates:

Use the Dynatrace synthetic monitoring API to manage synthetic monitors, including ping tests. You can create, update, or delete synthetic monitors based on your requirements.

ServiceNow Integration:

Use ServiceNow APIs to automate ticket creation, updates, or any other actions necessary for your workflow. This could involve creating an incident for decommissioning, agentless monitoring requests, or synthetic ping updates.

Orchestration and Automation:

Develop scripts or workflows to orchestrate the interactions between Dynatrace and DXC ServiceNow. This might involve using tools like PowerShell, Python, or integration platforms such as Zapier or Microsoft Power Automate.

Testing and Monitoring:

Test your integration thoroughly to ensure that decommissioning, agentless monitoring, and synthetic ping updates are working as expected. Monitor the integration for any errors or issues and refine as necessary.

Note\*\*\*Need credentials for Config API & a Role from the EKAM server\*\*\*\*

The "out-of-the-box" SNOW API is apparently the IRE API and an example of updating the install status of a CI follows:

URL:

https://api.stage.eu.platform.dxc.com/con4-api/dxc/configuration-items-ire/R1/ci

Body:

{

"items": [

{

"className": "<<serverClass>>",

"values": {

"correlation\_id": "<<correlationId>>",

"discovery\_source": "<<discoverySource>>",

"fqdn": "<<fqdn>>",

"install\_status": "Retired",

"name": "<<serverName>>",

"serial\_number": "<<serialNumber>>",

"sys\_id": "<<systemId>>"

},

"display\_values": {

"company": "<<company>>"

}

}

]

}

Headers:

Accept:application/json

Context-Type:application/x-www-form-urlencoded

Authorization:Basic <<encodedCreds>>

x-dxc-inf-route-key:<<discoverySource>>

x-dxc-inf-customer:<<company>>

Content Type:

application/json

Character Set:

UTF-8

Method:

POST

Notes:

“x-dxc-inf” settings probably don’t apply, since these are API gateway related, but there may be something similar in your environment

serverClass - e.g. cmdb\_ci\_linux\_server

correlationId - same as Dynatrace uses

discoverySource - probably the Dynatrace IRE API user

encodedCreds - Base64 encoding of "user:password" string. You may need to append "=".

proxy details may be necessary – they are in our customer, but then we are accessing the DXC SNOW.

It may be an idea to see if existing CIs can be queried before trying to update them. The method would be GET and an example URL would be:

https://api.stage.eu.platform.dxc.com/con4-api/dxc/configuration-items-ire/R1/ci/get?sys\_id=<<systemId>>

As I mentioned earlier, there is probably a tag that is used by the CI “pull” task that creates new SNOW CIs for servers enrolled in Dynatrace. In our customer, that tag is called “CMDB Filtered”. If the server is decommissioned before this tag is removed from the entity, duplicate CIs can be created. In our flows we have a check that waits for the tag to automatically disappear before we shut the server down. We also add tags to any servers that are decommissioned/due to be decommissioned (key = CIStatus, value = ”InDisposition” or “Retired”) – entities with these tags are excluded from the CI “pull” task.

## Maintenance Window API

To create a maintenance window in Dynatrace via ServiceNow, you'll need to utilize Dynatrace's REST API for maintenance windows and integrate it with ServiceNow. Here's a general outline of the steps you'd take:

Authentication Setup:

Obtain API tokens or credentials from both Dynatrace and ServiceNow to authenticate your requests.

Maintenance Window Configuration:

Use Dynatrace's REST API to create, update, or delete maintenance windows. You'll need to provide details such as the scope (e.g., specific hosts, services, or tags), timeframe, and purpose of the maintenance window.

ServiceNow Integration:

Utilize ServiceNow APIs to trigger the creation of maintenance windows in Dynatrace based on events or requests logged in ServiceNow. This could involve creating a custom ServiceNow application or workflow that interacts with both platforms.

# CONCEPTUAL Design

## Current CONCEPTUAL Design

A diagram of a service

Description automatically generated with low confidence

Figure 3: Current State

## Target CONCEPTUAL Design

A diagram of a service

Description automatically generated

Figure 4: Future State

# DXC Edge to DXC Platform X Data Classification

DXC Edge is deployed on premise with communication back to DXC Core Services using the DXC Private Network (SSI)

All data to and from the Platform X AWS CORE and the EDGE is encrypted via the Platform X API Gateway (HTTPS 443)

All communication between Platform X AWS CORE and Platform X ServiceNow (ITSM) is encrypted using the Platform X API Gateway.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DXC Edge Component | Direction | DXC Platform X | Data Type / Classification | Description |
| EKAM Raffia | **<<-->>** | API Gateway | Runbook automation tasks | Automation requests and responses via the automation API on encrypted HTTP 443 |
| EKAM Raffia | **<<-->>** | Master EKAM | Install Media Sync (REPO) | Edge X 2.0 Binaries are pulled form Master using Ports 32002, 9292 which are secured using SSL/TLS 1.2 |
| ActiveGate(s) | **<<-->>** | Dynatrace Cluster | Monitoring Policy and Event Data | Certificate based communication on port 443 utilizing X.509 encryption standards. Events are sent to Dynatrace Cluster (DXC CORE) where they are correlated/deduped. |
| Dynatrace Cluster | **-->>** | ITOM Event / Incident | Event Data | Events sent to Dynatrace Cluster, are forwarded to ServiceNow for remediation via manual or automated response. |

**Table 1: Data Classification**

# Assumptions, Risks and Dependencies

## Assumptions

| Assumptions ID | Description |
| --- | --- |
| A01 | Only DXC personnel will have access to Platform X – Edge X 2.0 services |
| A02 | Compute resources are available in the Lake Park and Tonaquint Enterprise Compute to provide the requested virtual machines. |
| A03 | Capacity exists on the appropriate Shared Storage (SAN) devices to provide the requested storage sizing to support the project delivery. |
| A04 | Any IM environment change freeze will impact implementation timeline. |

Table 2: Assumptions

## Risk Register

| Criticality | Risk | Mitigation |
| --- | --- | --- |
| Low | If network latency is an issue between managed servers and satellite servers then additional satellite servers will be required | Ensure enough testing is done to determine latency impact. |

Table 3: Risk Register

## Dependencies

| Dependency ID | Description | Owner |
| --- | --- | --- |
| D01 | There is sufficient computing resource at Lake Park and Tonaquint DC to host the new virtual machines. | Intermountain - DXC |
| D02 | Intermountain Security to review, approve the major upgrade and provide the evidence of approval. | Intermountain |
| D03 | Intermountain Architecture to review, approve and provide the evidence of approval. | Intermountain |

Table 4: external dependencies

Appendix A Control

* 1. Document Authorization

| Name | Role | Date |
| --- | --- | --- |
| Satyanarayana VeeraVenkata (Satya) Boddu | Project Lead | 4/28/2024 |
| James (Jim) Hall | Principle Author | 4/28/2024 |

Table 5: Document Authorization

* 1. Document Distribution

| Name | Role | Date |
| --- | --- | --- |
| Swetha Bakthavatsalan | Technical Peer Review | 5/2/2024 |
| Terese Patel | Product Peer Review | 5/2/2024 |
| Russ Diller | Platform Architect – Peer Review | 5/2/2024 |
|  |  |  |
|  |  |  |

Table 6: Document Distribution

Change History

| Version | Date | Author | Summary of changes |
| --- | --- | --- | --- |
| 0.1 | 4/23/2024 | James (Jim) Hall | First draft |
|  |  |  |  |

Table 7: Change History

* 1. Document References

| Document | File Name |
| --- | --- |
|  |  |
|  |  |
|  |  |

Table 8: Document References

* 1. Exception Request Approvals

| Exception ID | Exception Details | Status |
| --- | --- | --- |
| N/A | None required |  |

Table 9: Exception Request Approvals

1. Glossary of Terms

| **Reference** | **Description** |
| --- | --- |
| AD | Active Directory |
| AED | Architectural Engagement Document |
| AT&T | American Telephone and Telegraph. Network provider for DXC Intermountain Account |
| AIX | IBM’s Advanced Interactive eXecutive Unix based operating system |
| AM3 | DXC MFSA Americas 3 Mesh. US based implementation of DXC MFSA |
| C2SSN | Client to DXC SSN Firewall |
| CA | Certificate Authority |
| CIFS | Common Internet Filesystem |
| CM | Configuration Management |
| CMDB | Configuration Management Database |
| COTS | Commercial Off-The-Shelf |
| CPU | Central Processing Unit |
| CSV | Comma-Separated Values |
| CoD | Capacity on Demand |
| CORE Facility | See MESH |
| CORE | Individual server within a Mesh |
| DB | Database |
| DC | Data Center |
| DDR | Double Data Rate |
| DFS | Distributed File System |
| DHCP | Dynamic Host Configuration Protocol |
| DNS | Domain Name System |
| DOS | Disk Operating System |
| DR | Disaster Recovery |
| DoS | Denial of Service |
| ESX | VMware virtualization host |
| FQDN | Fully Qualified Domain Name |
| FTP | File Transfer Protocol |
| GB | Gigabyte |
| GRAS | Global Remote Access Service |
| Gb | Gigabit |
| Gbps | Gigabits per second |
| HBA | Host Bus Adapter |
| HDD | Hard Disk Drive |
| HPUX | HPE Unix operating system |
| HTML | Hypertext Markup Language |
| HTTP | Hypertext Transport Protocol |
| Hz | Hertz |
| IM | Intermountain Healthcare Care |
| INFRASEC | DXC internal tool for managing SSI security |
| IP | Internet Protocol |
| ISL | Infrastructure Service Line |
| ISO | International Organization for Standardization |
| ISP | Internet Service Provider |
| ISV | Independent Software Vendor |
| LDAP | Lightweight Directory Access Protocol |
| LP | IM Lake Park data center |
| MACD | AT&T routing request change |
| MFA | Multi Factor Authentication |
| Mesh | A group of MFSA-Core-Servers that work together for scalability and redundancy |
| NAT | Network Address Translation |
| NetSRM | Network Service Request |
| OVA | Open Virtual Appliance |
| SELINUX | Security-Enhanced Linux |
| SGW | DXC Infrasec Secure Gateway |
| SSH | Secure Shell |
| SSI | DXC Shared Services Interconnect |
| SSN | DXC Secure Services Network |
| UDI | Universal Discovery Inventory |
| USR | Urban Southern Region |

Table 10: Glossary of Terms