Configuration Management

- The goal of configuration management is to ensure that accurate and meaningful information is readily available about the configuration of applications and services and the configuration items (CI) that support them
- Includes all relationships and dependencies between the CIs
- Objects include hardware, software, networks, sites, vendors, suppliers, and people

Configuration Management

- Configuration management (CM) is a governance and systems life cycle process for ensuring consistency among all assets (configuration items, or CIs) in an operational environment
- Classifies and tracks individual CIs
- Documents functional capabilities and interdependencies
- Verifies the effect a change to one configuration item has on other systems
- There are key differences between change management and configuration management – configuration management comes first

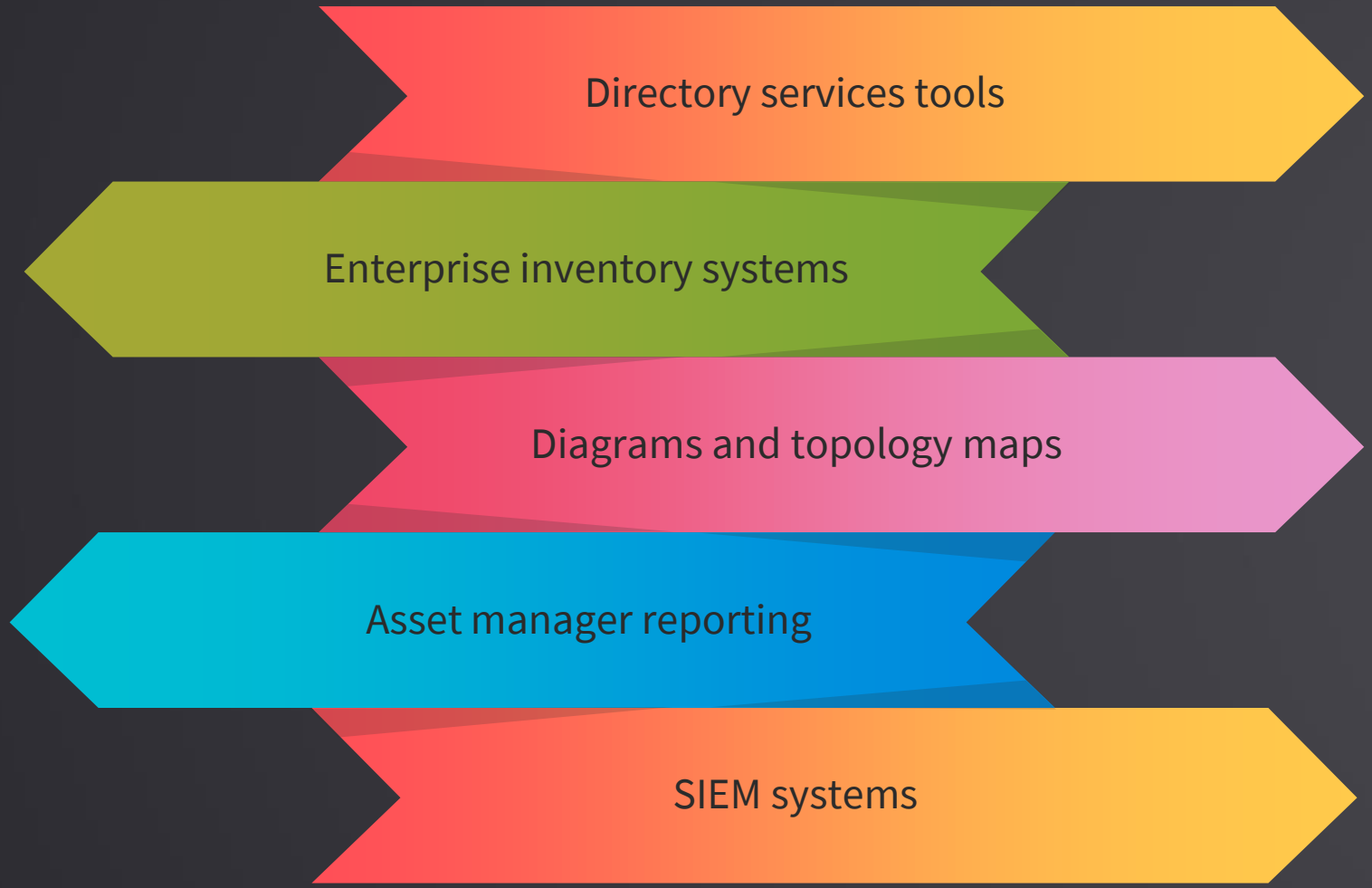
Configuration Management Database

- A configuration management system (CMS) is a set of data, tools, utilities, and processes used to support configuration management
- All information should be tagged and labeled with a common unified schema, preferably using key-value pairs
- This data will populate a database system known as a configuration management database (CMDB)
  - Relational databases have been used historically
  - NoSQL/document databases are emerging as a common solution
  - Could leverage a CSP service, such as Amazon DynamoDB

Configuration Management Database

- Plays a critical role in several IT management initiatives, like IT service management (ITSM) and IT asset management (ITAM)
- Helps various IT services to better align with business needs by providing current and accurate data for:
  - Change and patch management
  - Incident and problem management
  - Availability management
  - Release and deployment management

Configuration Management Sources



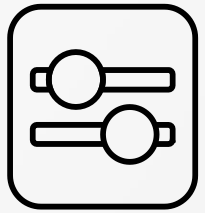
Change Management

- Change management is also called the "change control practice"
- The goal is to maximize the amount of successful service and product changes
- Should make certain that risks have been adequately assessed, authorized, and managed with a change schedule
- Operates with the configuration database to track all possible dependencies and repercussions of changes
- Involves a change log or change database

Types of Changes

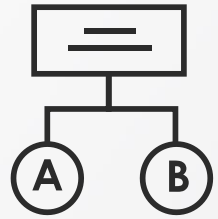
Standard

- Low-risk changes
- Pre-authorized and well-documented
- Can be automated
- Service requests that don't need additional authorization
- Example: changing directory password



Normal

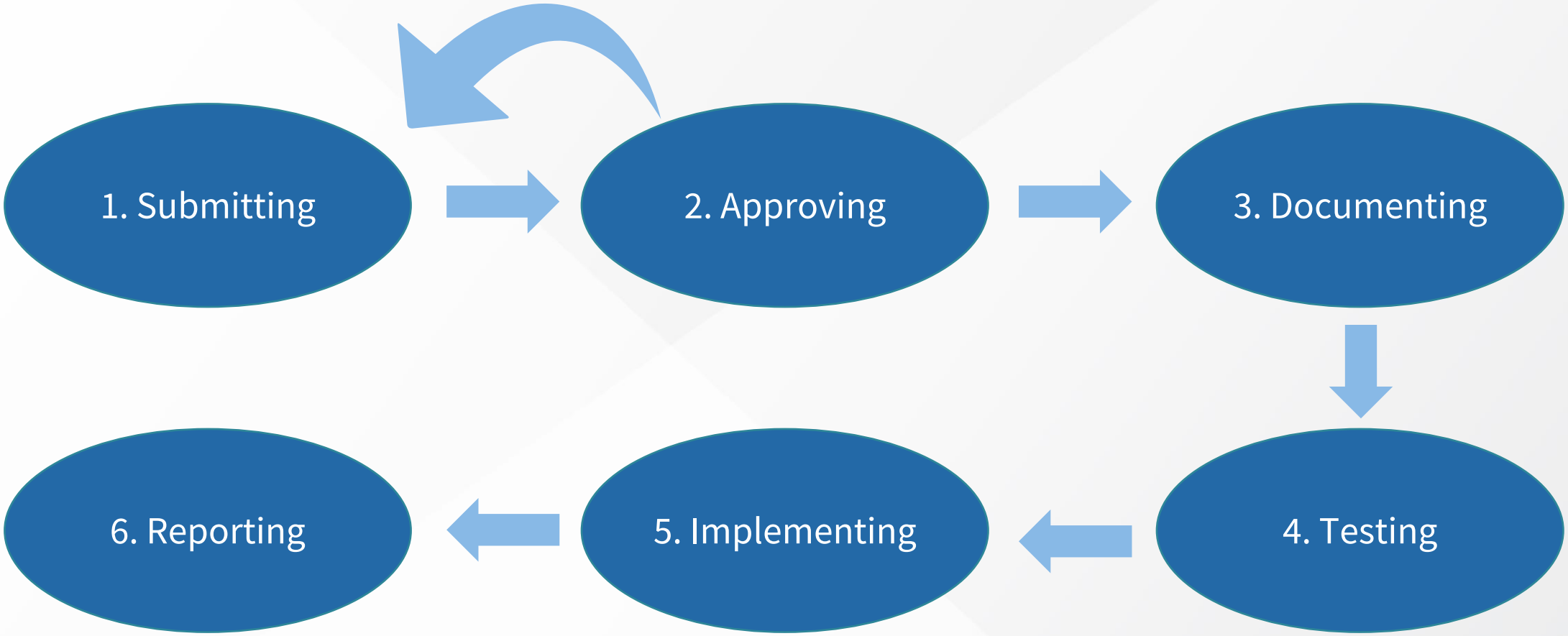
- Changes follow a specific process for scheduling, assessment, and authorization
- Are lower risk, but do go through an approval process
- Example: onboarding a new phone or laptop; installing an application



Emergency

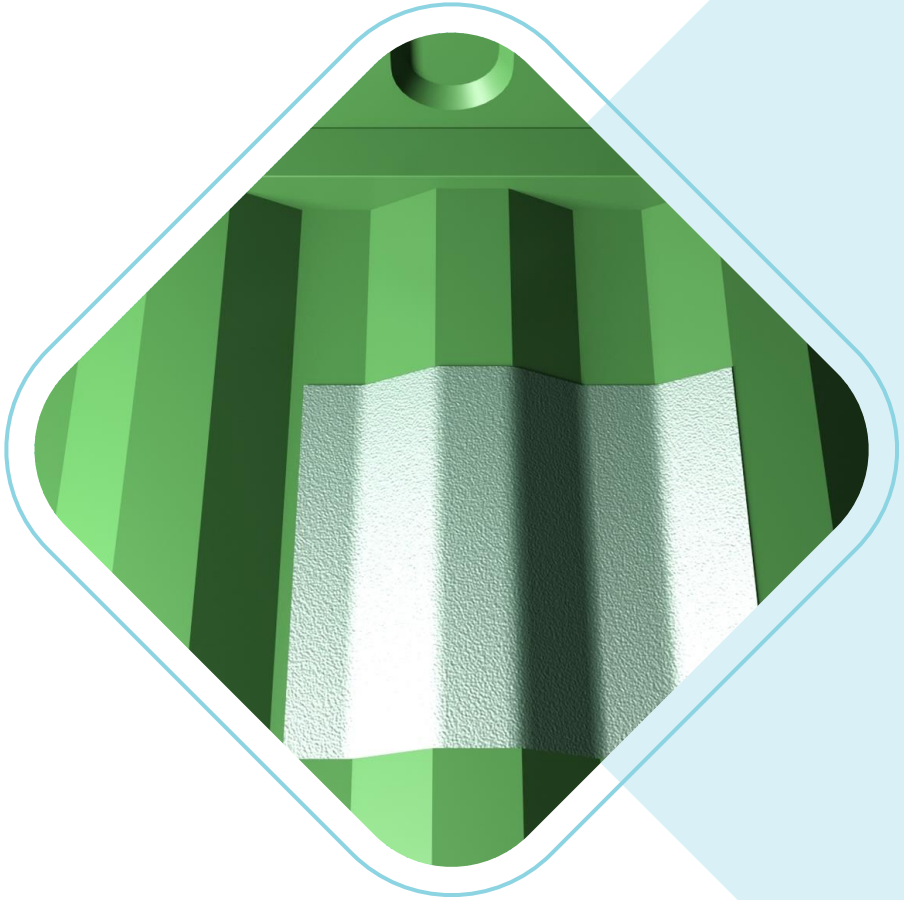
- Changes must be implemented immediately
- Often a result of problem management or after-action reporting
- May involve escalation or an emergency advisory board if the amount of resources or disruption is significant

Change Management Lifecycle

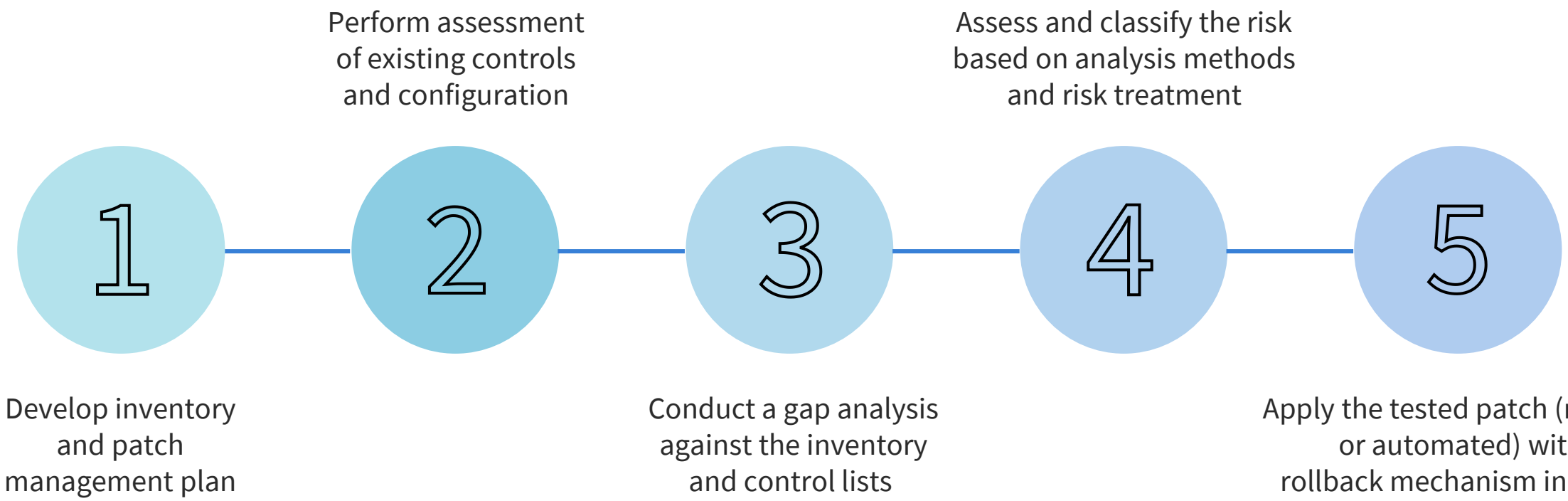


Patch Management

- Many organizations do not consider or continually improve their patch management plan
- Vulnerability/exposure reviews and gap analysis are not performed or done properly
- Patch management should compliment configuration and change management
- Only certain personnel should have the authority to test, apply, and determine the urgency of patching activities
- Agreements with any applicable vendors should also be made to address any potential issues before patch deployment



Patch Management



Continuity Management

- The purpose of the service continuity management practice is to ensure that the availability and performance of a service is maintained at a sufficient level in the event of a disaster
  - A disaster is a sudden unplanned event that causes great damage or serious loss to an organization
  - It results in an organization failing to provide critical business functions for some predetermined minimum period of time

Service Continuity Management

This practice offers a framework for building organizational resilience with these goals

Key stakeholder interests

Organizational reputation

Value-creating activities

Corporate brand and image

Business Impact Examples

Disaster sources	Stakeholders involved	Organizational impact
Supply chain failure	Employees	Lost income
Terrorism	Executives	Damaged reputation
Weather	Governing body	Loss of competitive advantage
Cyberattack	Suppliers	Breach of law, health and safety regulations
Health emergency	IT teams	Risk to personal safety
Political or economic	Customers	Immediate and long-term loss of market share
Technology failure	Users	
Public crisis	Communities	

Continuity vs. Incident Management

Continuity management focuses on significant events that can lead to disaster or catastrophic disruption of organizational activities

Incident management focuses on reducing the immediate impact of lesser negative occurrences

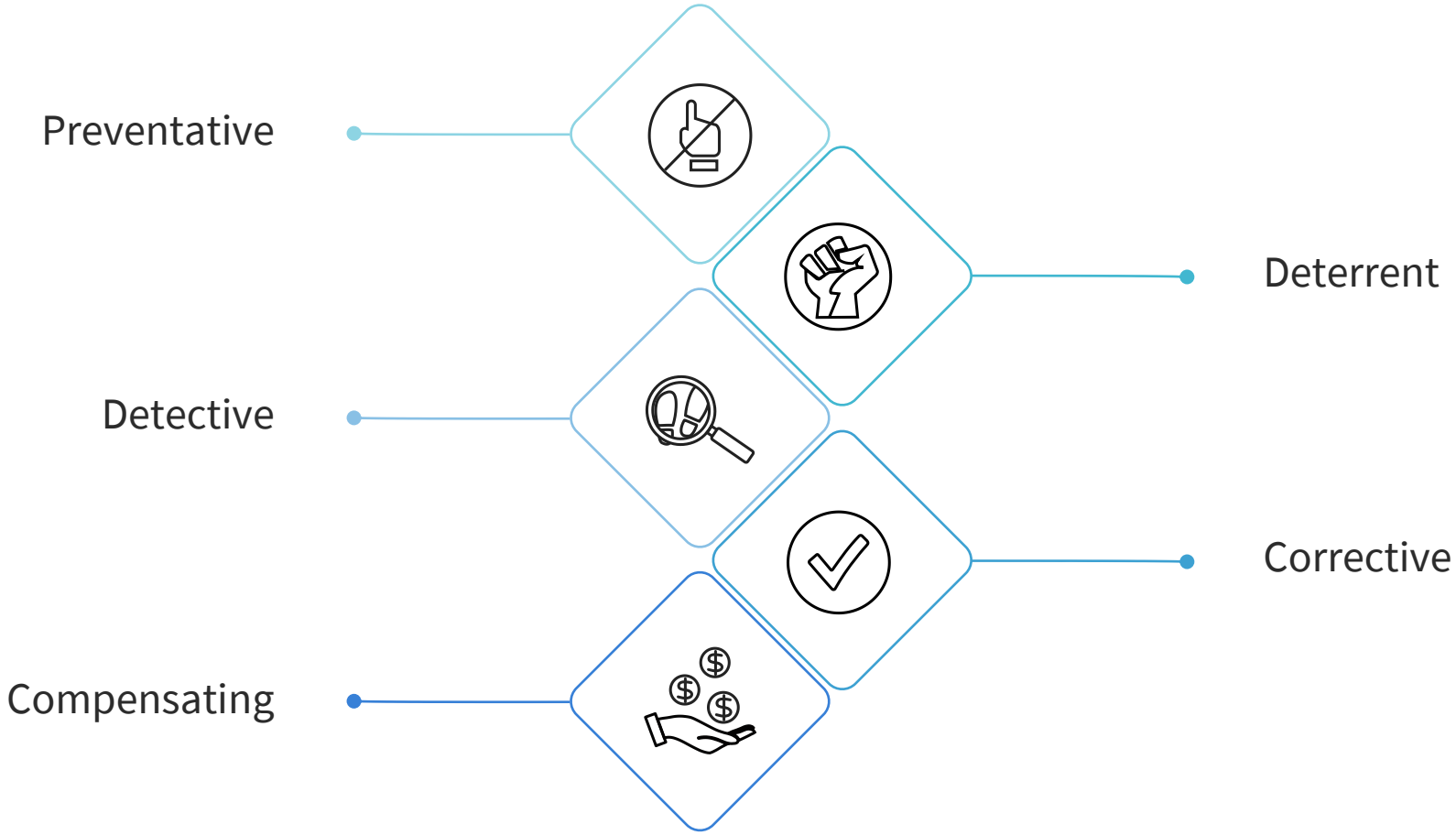
Information Security Management

- The purpose of the information security management practice is to protect the information needed by the organization to conduct its business by delivering:
  - Confidentiality
  - Integrity
  - Availability
  - Authentication
  - Non-repudiation

Categories of Controls

- Administrative controls (managerial)
- Technical and operational controls
- Physical controls

Types of Controls



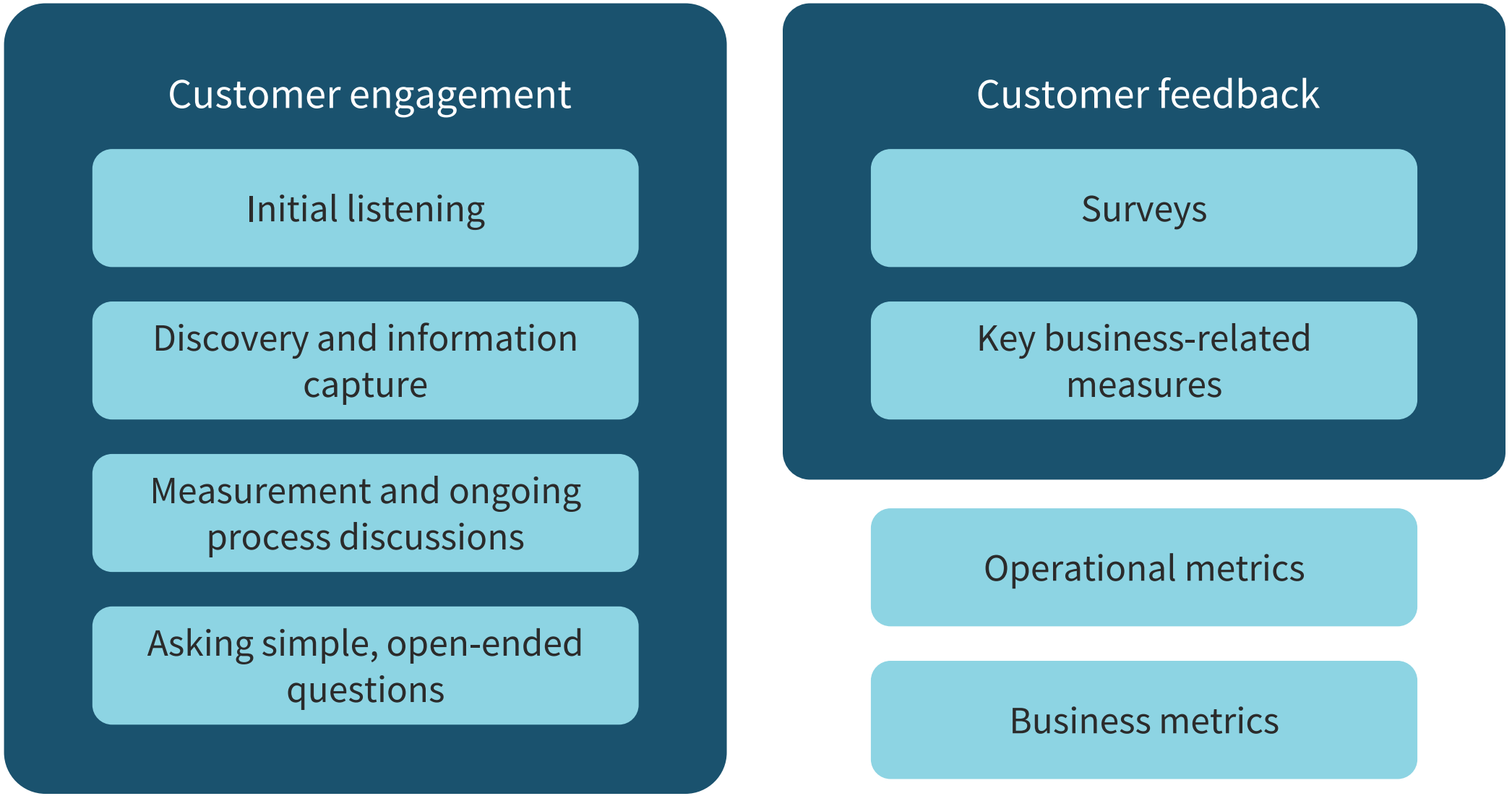
InfoSec Management: Processes and Procedures

- An information security incident management process
- A risk management process
- A control review and audit process
- An identity and access management process
- Event management
- Procedures for pentesting, vulnerability scanning, etc.
- Procedures for managing information security-related changes

Service Level Management

- The purpose of the service level management practice is to set clear business-based targets for service performance so that the delivery of a service can be properly assessed, monitored, and managed against these targets:
  - Provides the end-to-end visibility of the organization's services
  - Establishes a shared view of the services and target service levels with customers
  - Collects, analyzes, stores, and reports relevant metrics to ensure service levels are met
  - Performs service reviews, captures, and reports on service issues, including performance against defined service level

Service Level Management Information Sources



Service Level Agreements (SLAs)

- A service level agreement (SLA) is a documented agreement between a service provider and a customer that identifies services required and expectations
- It is a tool to measure the performance of services from the customer's point of view
- Requirements for a successful SLA:
  - Should relate to a defined service
  - Should pertain to defined outcomes, not just operation metrics
  - Should reflect an agreement between the service provider and the service consumer
  - Should be simply written and easy to understand for all parties

Service Request Management

- Service requests are pre-defined and pre-agreed and can usually be formalized with clear, standard procedures
- The purpose of service request management is to support the quality of a service by handling all user-initiated service requests in an effective and user-friendly manner
- A service request is a request from a user or a user's authorized representative that initiates a service action that is part of normal part of service delivery
- Service requests are a normal part of service delivery, not a failure or degradation of service, which are handled as incidents

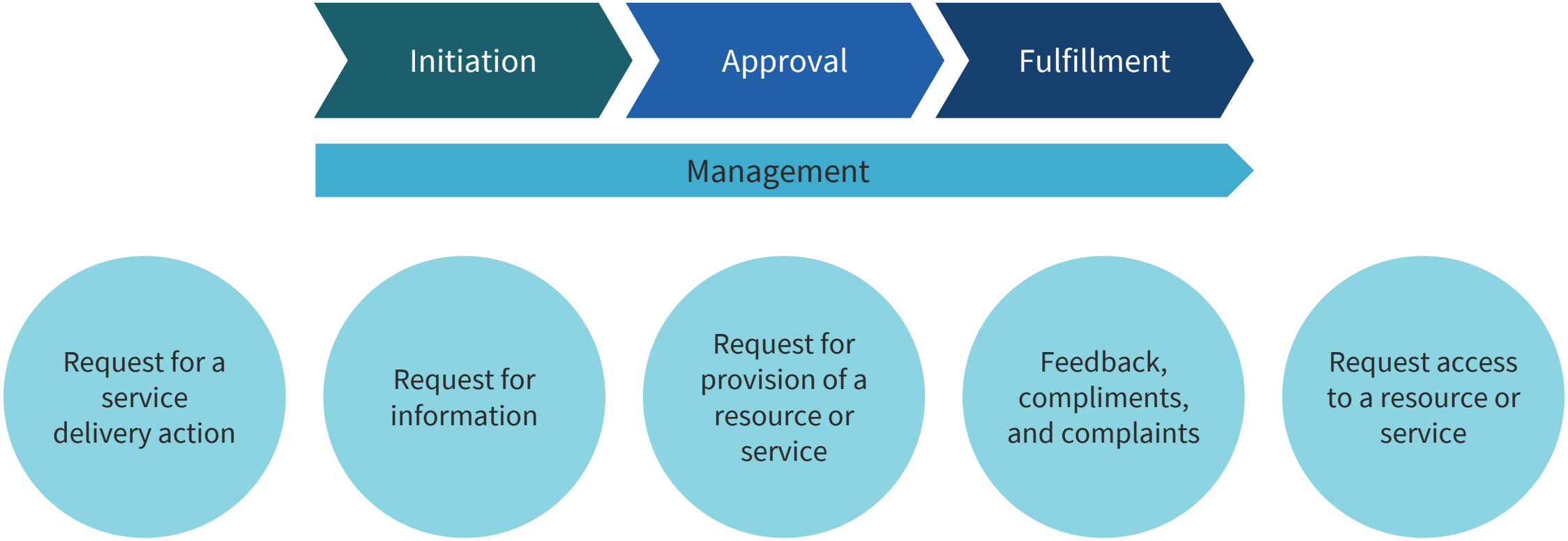
Service Request Management

- Service requests and their fulfilment should be standardized and automated to the greatest degree possible
- Policies should be established regarding what service requests will be fulfilled with limited or even no additional approvals so that fulfilment can be streamlined
- Policies and workflows are needed to redirect service requests that should be managed as incidents or changes

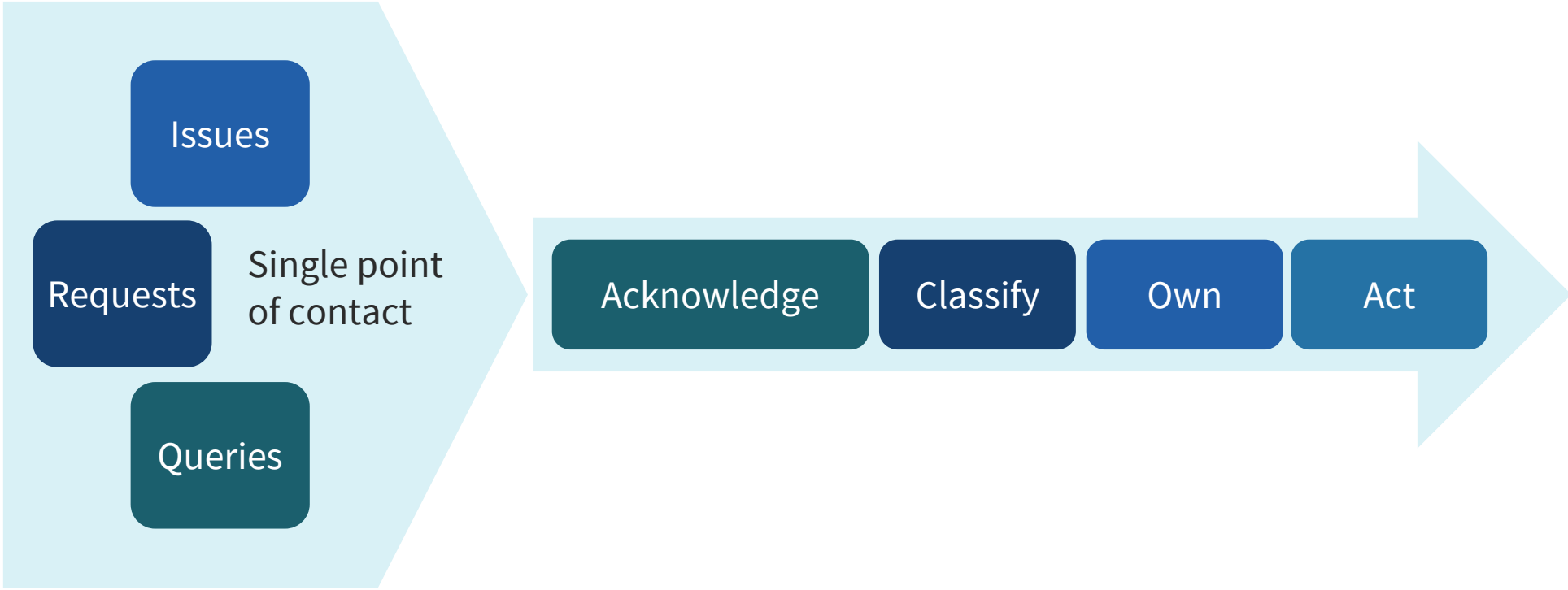
Service Request Management

- Opportunities for improvement should be identified and implemented to produce faster fulfilment times and take additional advantage of automation
- The expectations of users regarding fulfillment times should be clearly set, based on what the organization can realistically deliver
- Some service requests require authorization according to financial, information security, or other policies
- Service request management depends on well-designed processes and procedures, which are operationalized through tracking and automation tools

Service Request Management

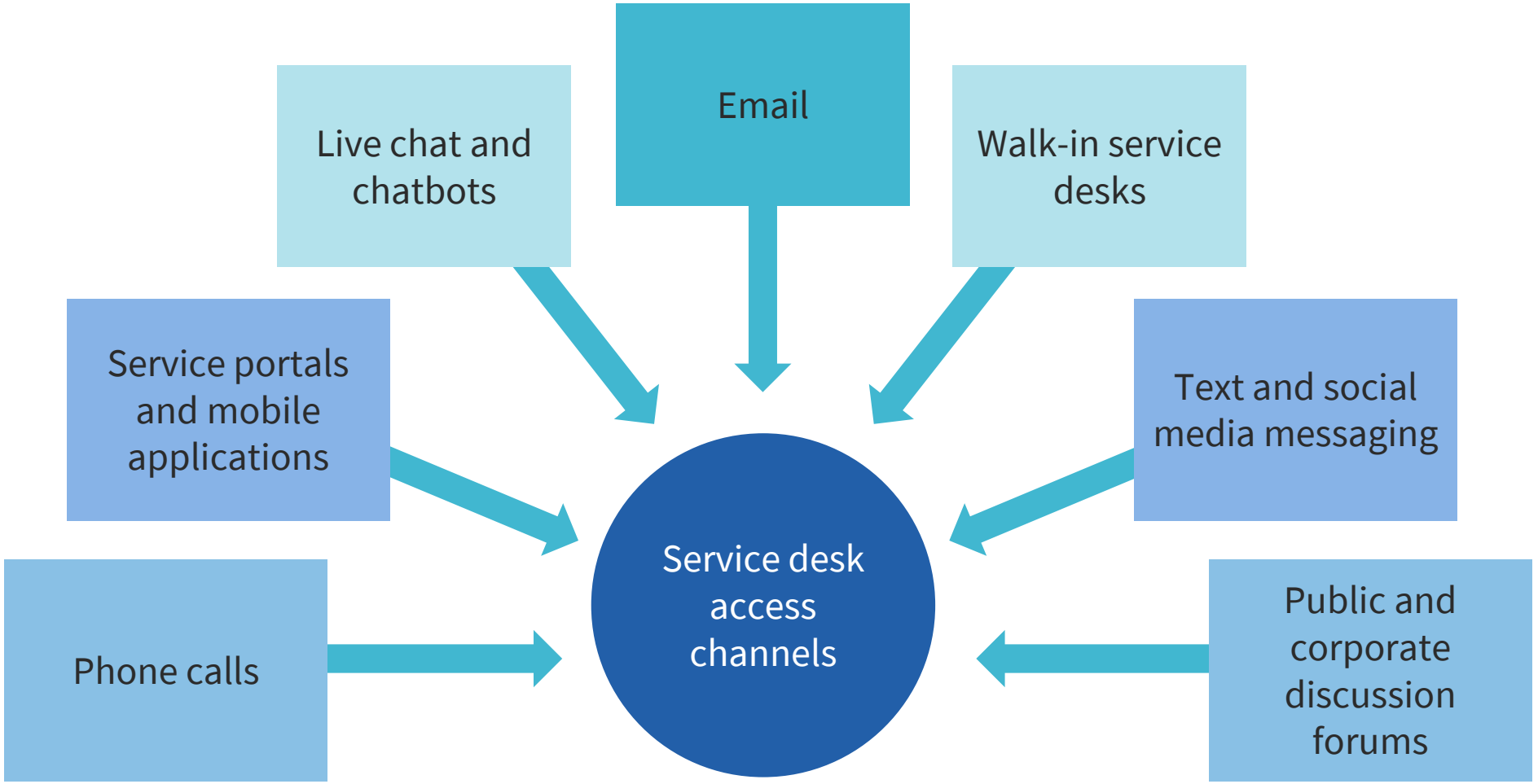


Service Desk: Process



Service Desk: Skillset

Service Desk: Access Channels



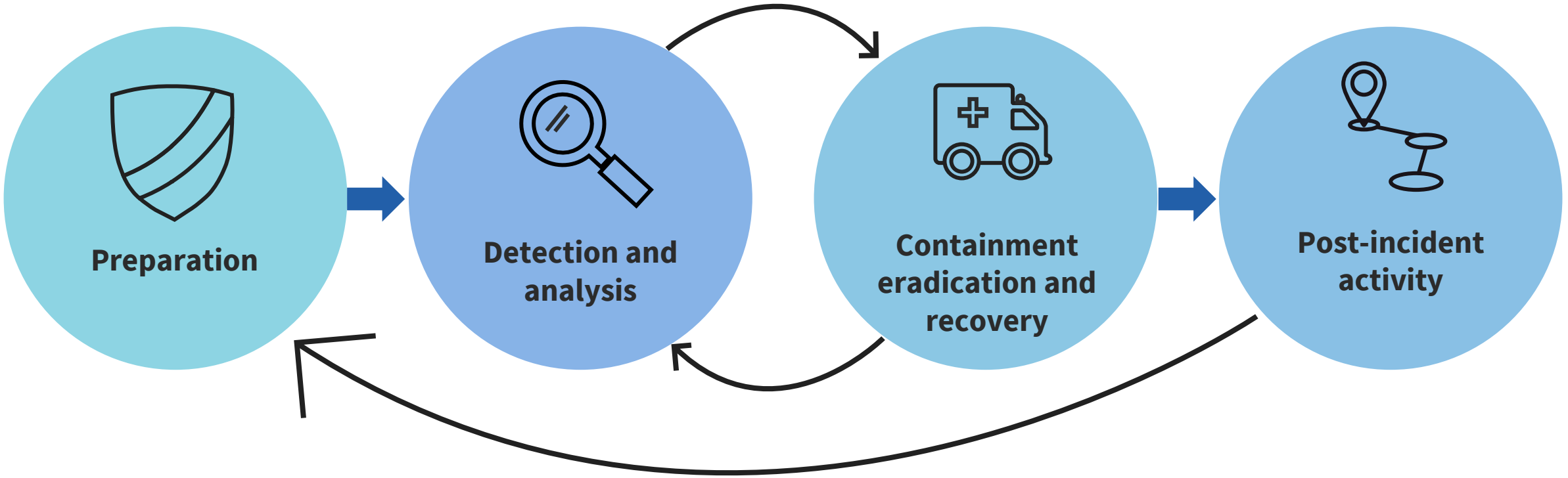
Incident Management and Response

- Steps taken when a negative event disrupts normal operations
- Primary goal is to reduce the immediate impact
- Should have documented incident types/category definitions based on risk assessments, risk registers, and business impact analysis (BIA)

Incident Management and Response

- Know roles and responsibilities of the first responders, including reporting requirements and escalation processes
- Collect contact lists, contact public relations people and legal teams
- Best practice is to have pre-performed exercises, drills, and simulations

Cyber Incident Response Cycle



Problem Management

- A problem is the root cause or potential source of one or more negative occurrences
- Problems can originate from major incidents affecting many users or from repeated incidents
- Problems can further be recognized in infrastructure diagnostic systems (SIEM) before users are affected
- Problem management is a systematic method to warrant that minimal incidents arise from IT infrastructure operations by probing deep into all relevant events to find the root causes and fixes

Problem Management

- Effective problem management can also reduce the severity of the incidents through suitable documentation of existing issues and providing workarounds
- It is a methodical approach to identify the cause of an incident and manage the life cycle of all problems
- The goal is to minimize the impact of incidents and eliminate recurring ones

Problem Management: Three Phases

1. Problem identification

Identify the problem and log it in a problem management tool

2. Problem control

Prioritize, investigate, and analyze the logged problems

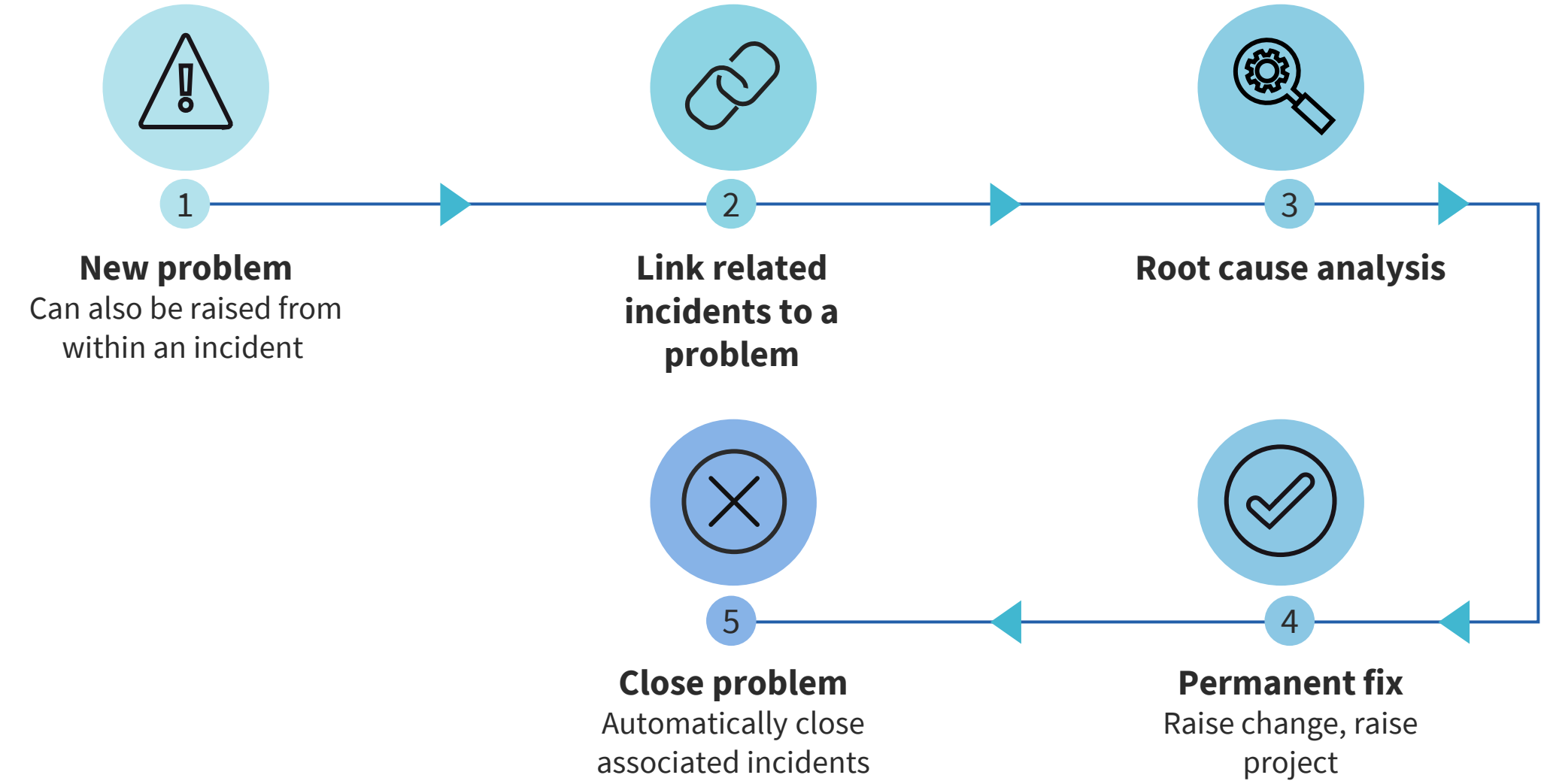
3. Error control

Manage known errors from the KEDB periodically

Problem Management: Roles and Responsibilities

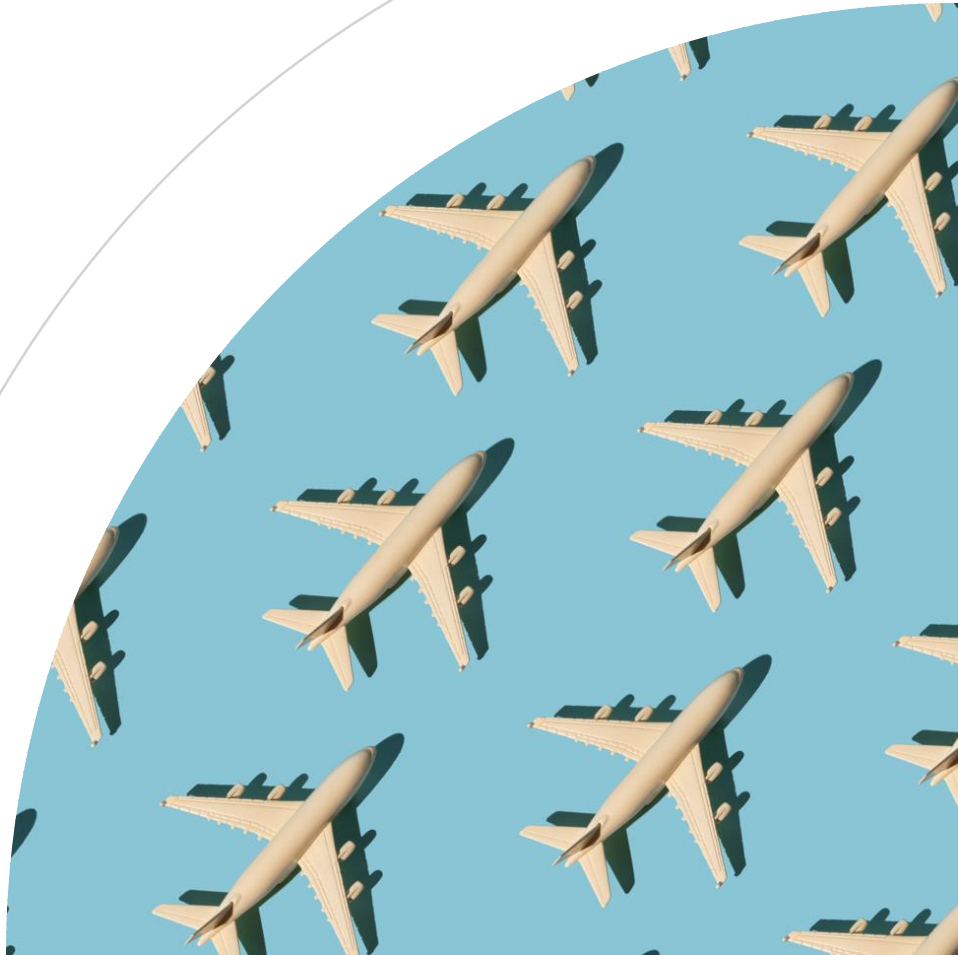
Role	Responsibility
Problem manager	Responsible for the effectiveness and efficiency of the entire practice, akin to team leader
Problem owner	Accountable for the cycle of any problem tickets they're assigned
Problem agent	Accountable for the tasks associated within a problem ticket
Diagnosis team	An assortment of people with various expertise, responsible for RCA of a problem

Problem Management: Process

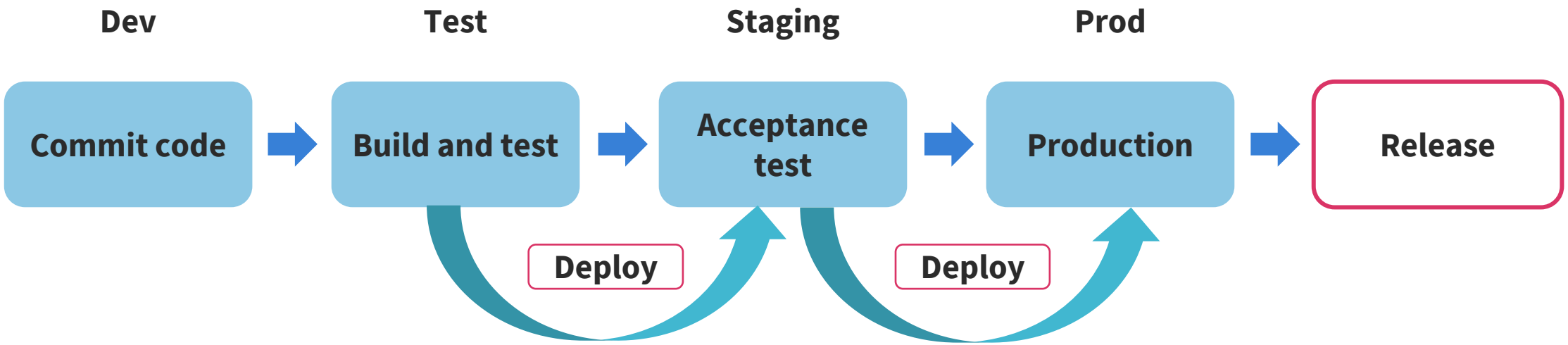


Release and Deployment Management

- The purpose of the deployment management practice is to move new or changed hardware, software, documentation, processes, or any other component to live environments
- It may also be involved in deploying components to other environments for testing or staging
- An example is staging a cloud-based disaster recovery solution like AWS CloudEndure



Release and Deployment Management



Managing Deployment Types

Phased deployment

New/changed components are deployed to just part of the production environment at a time (e.g., to users in one office or country) and is repeated as many times as needed until the deployment is complete

Continuous delivery

Components are integrated, tested, and deployed when they are needed, providing frequent opportunities for customer feedback loops

Big bang deployment

New/changed components are deployed to all targets at the same time, which is sometimes needed when dependencies prevent the simultaneous use of both the old and new components

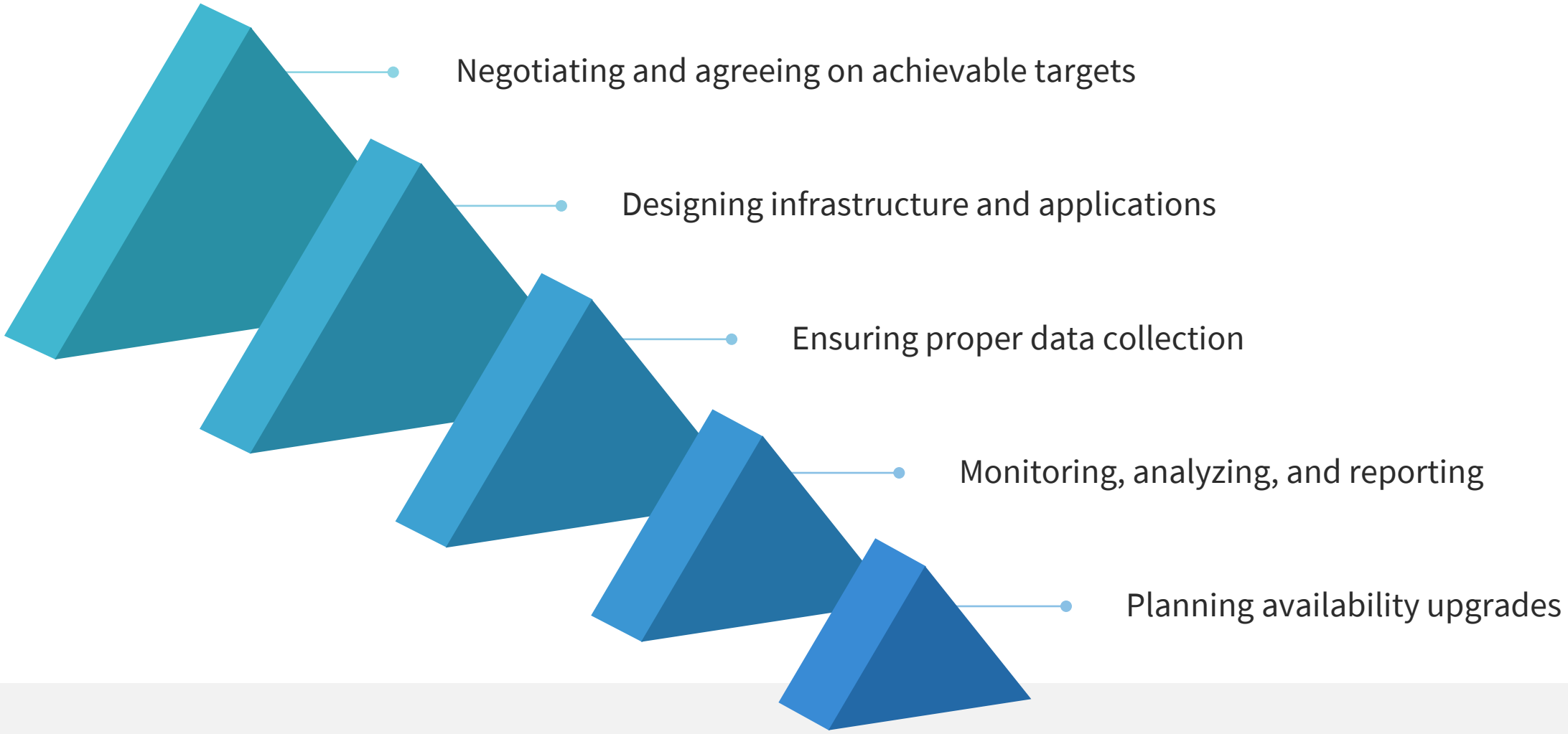
Pull deployment

New/changed software is made available in a controlled repository and users download the software to client devices when they choose, which allows users to control the timing of updates

Availability Management

- Availability is the ability of an IT service or other configuration item to perform its agreed-upon function when required
- The purpose of the availability management practice is to ensure services deliver agreed-upon levels of availability to meet the needs of customers and users

Availability Management Activities



- Negotiating and agreeing on achievable targets
- Designing infrastructure and applications
- Ensuring proper data collection
- Monitoring, analyzing, and reporting
- Planning availability upgrades

Availability Measurements

Mean time between failures (MTBF)

Mean time to restore service (MTRS)

User outage minutes

Number of lost transactions

Lost business value

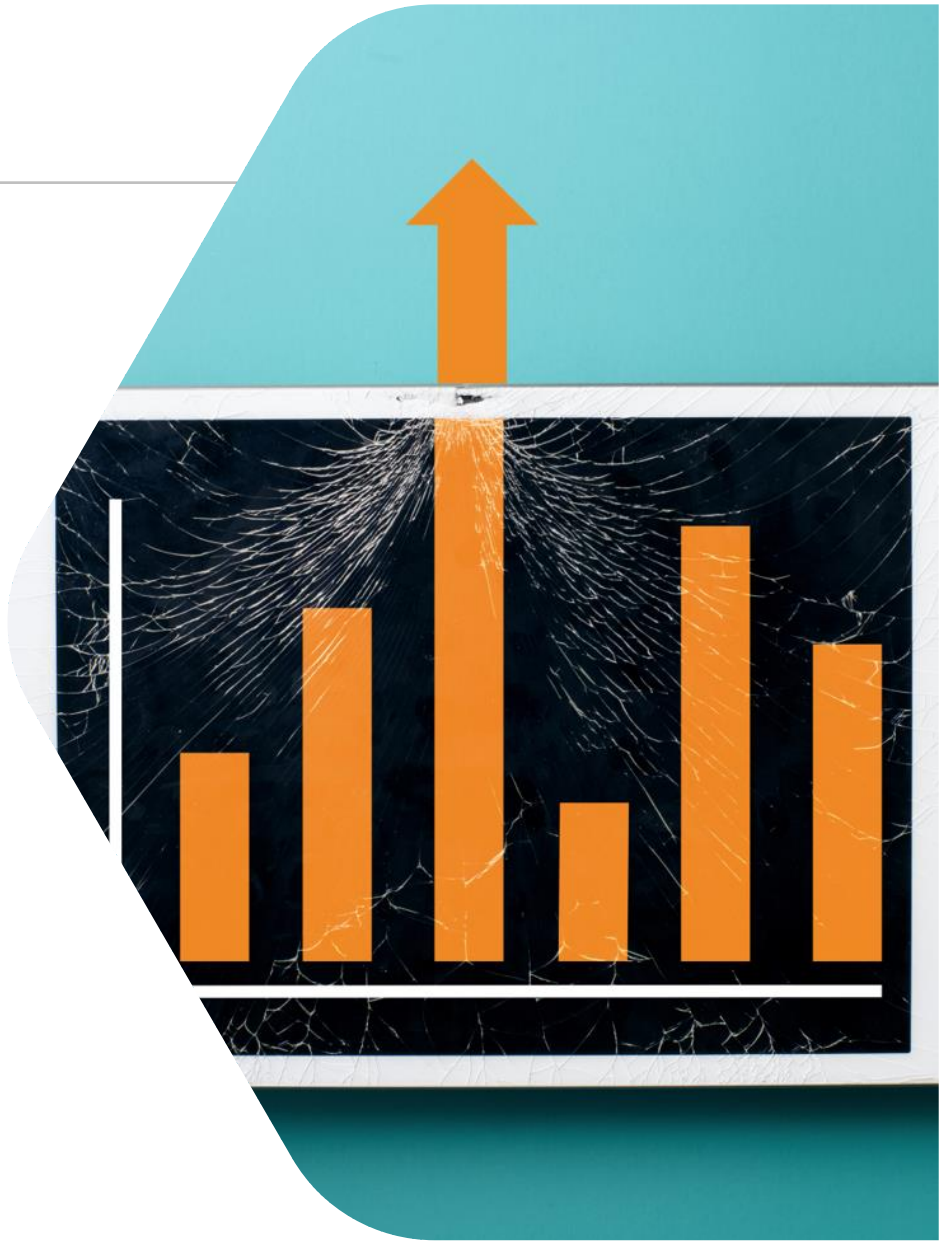
User satisfaction

Capacity Management

- The purpose of the capacity and performance management practice is to ensure that services achieve expected performance, satisfying current and future demand in a cost-effective way
- Performance is a measure of what is achieved or delivered by a system, person, team, practice, or service

Capacity Management

- Service performance is associated with the number of service actions performed in a timeframe and the time required to fulfill a service action at a given level of demand
- Service capacity is the maximum throughput that a configuration item or service can deliver



Capacity and Performance Management

- Researching and monitoring the current service performance
- Capacity and performance modeling
- Capacity requirements analysis
- Demand forecasting and resource planning
- Performance improvement planning