JOHNS HOPKINS
APPLIED PHYSICS LABORATORY

Engineering Design Patterns

The Johns Hopkins Applied Physics Laboratory

Brooke Guare
Cyber Systems Engineer
Brooke.Guare@jhuapl.edu

DISTRIBUTION STATEMENT A: Approved for public release. Distribution is unlimited.

Introduction

Background: In order to aid engineers in designing sufficiently cyber resilient systems, the Office of the Under Secretary of Defense for Research and Engineering (OUSD (R&E)) / Resilient Systems (RS) tasked the Johns Hopkins University Applied Physics Laboratory (JHU/APL) to curate and develop design patterns.

Challenge: The majority of systems have been designed to meet physical performance and functional requirements, as well as be resilient to a set of kinetic threats. However, there has not been as much attention paid to the resilience of the system to cyberspace threats.

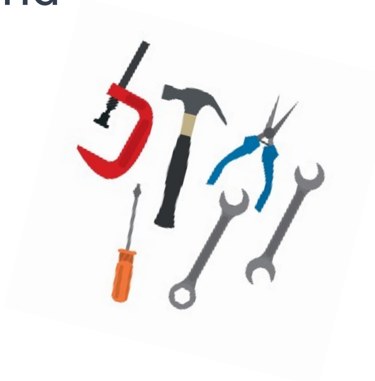
Approach

Solution: Development of design patterns

- A *design pattern* is a general, reusable solution to commonly occurring problems within a given context in system design

Impact: Compile design patterns proven successful or asserted to be useful, in order to:

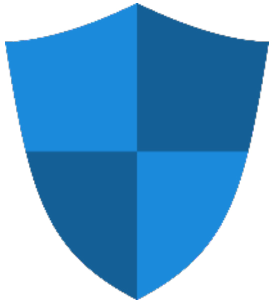
- Allow engineers to identify gaps and mitigate potential cyber related problems in their system
- Provide building blocks for cyber resilient system design
- Provide engineers the tools and knowledge they need to build resilient systems and meet cybersecurity requirements



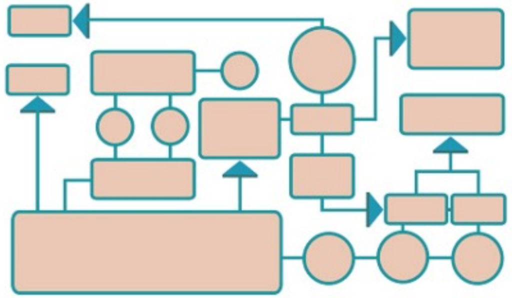
Cybersecurity-related Requirements

Design Patterns

+



Security Controls

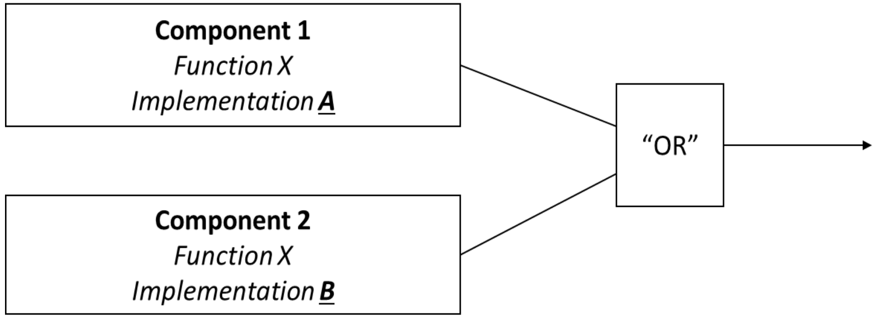


System Design

Resilient System

Case Study: Aircraft

Flight controls are electrically controlled



Component 1
Function X
Implementation ***A***

Component 2
Function X
Implementation ***B***

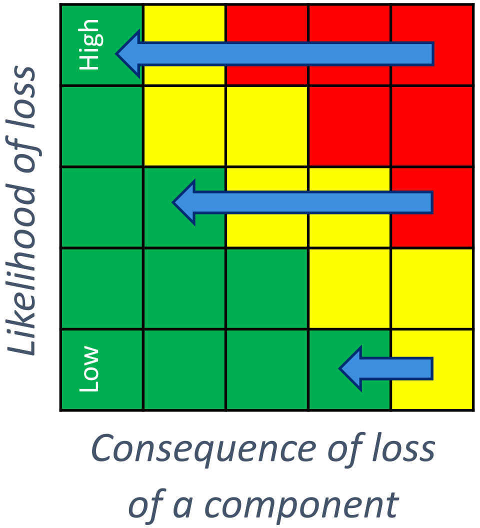
"OR"

Threat:

- Loss of power to mission critical components

Application of Diverse Redundancy Design Pattern:

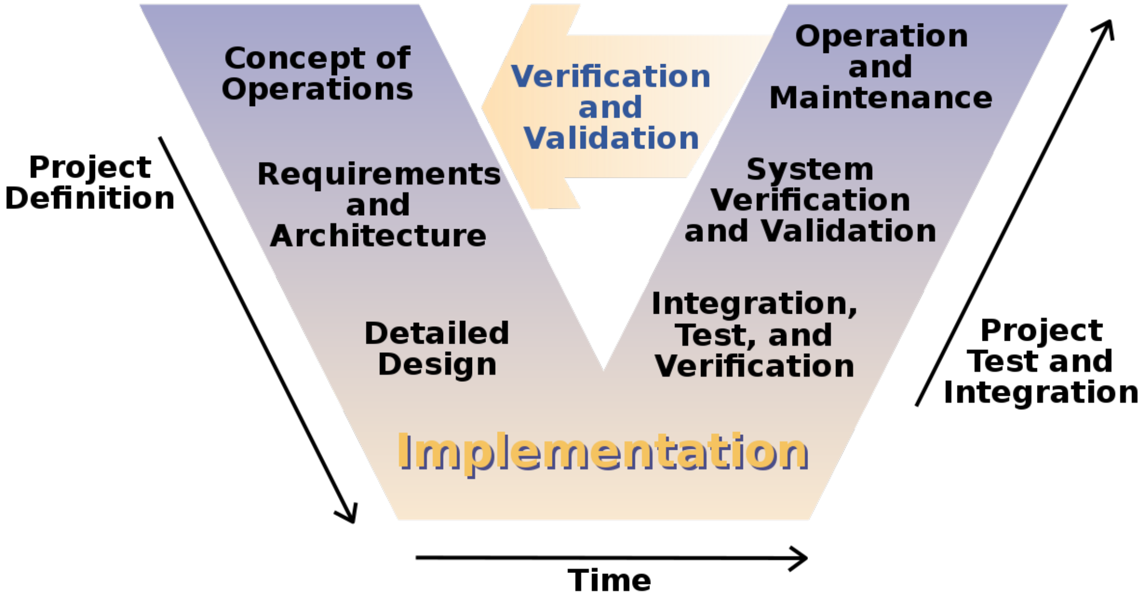
- Magnetic generator (primary source) allows power to be generated as long as engines are spinning
- 3 Electric Generators can power flight controls
- If electric backups fail, there is a battery backup



These mechanical examples can be translated to the cyber domain

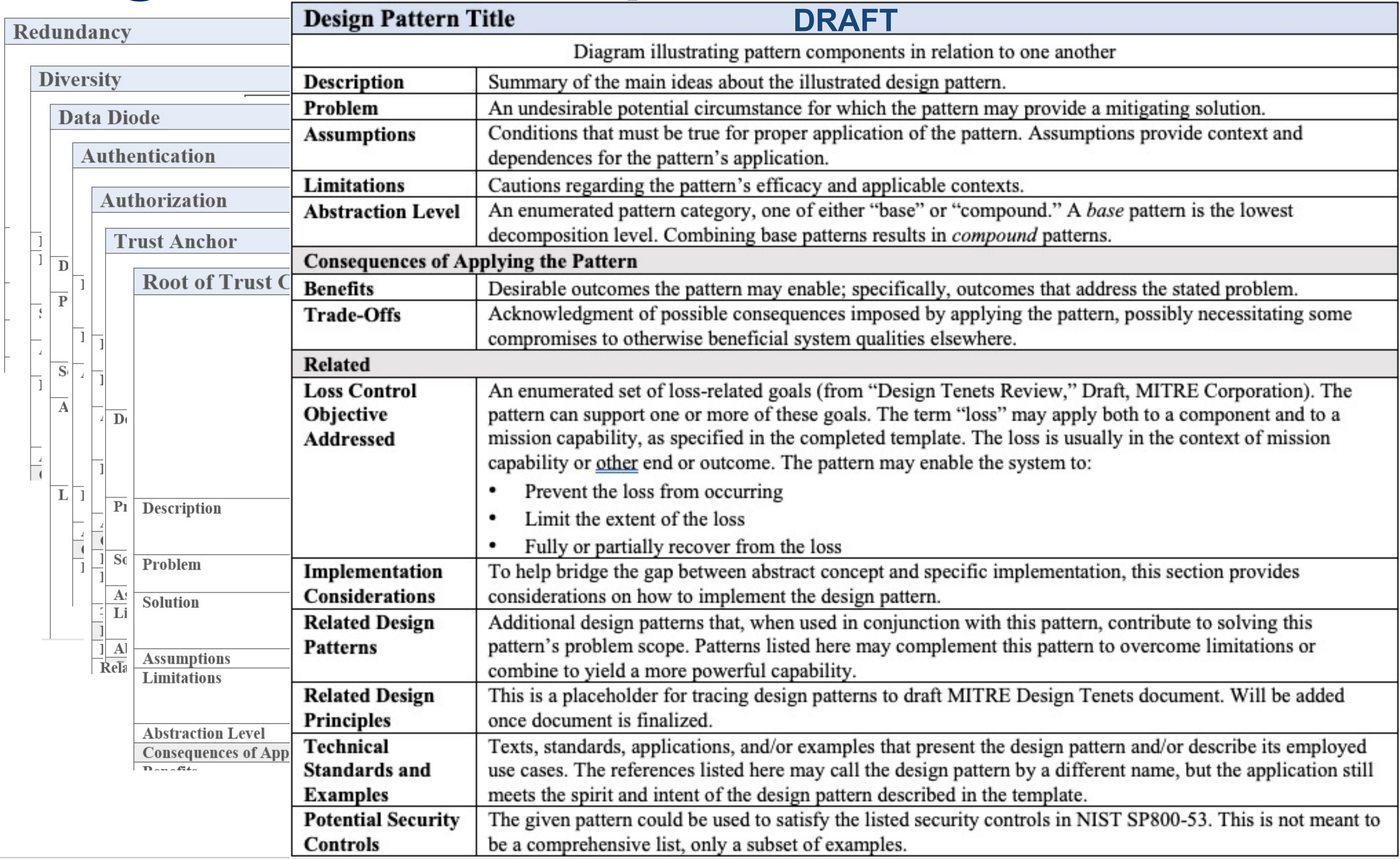
When Should Design Patterns Be Used?

- Integrating good design principles early in the systems engineering lifecycle helps ensure the system will be able to be resilient to the threat event, or set of threat events


Concept of Operations
Requirements and Architecture
Detailed Design
Implementation
Integration, Test, and Verification
System Verification and Validation
Operation and Maintenance
Verification and Validation
Project Definition
Project Test and Integration
Time

- However, design patterns can be applied throughout the systems engineering lifecycle in order to help secure existing systems

Design Pattern Template



Redundancy

Diversity

Data Diode

Authentication

Authorization

Trust Anchor

Root of Trust C

Description

Problem

Solution

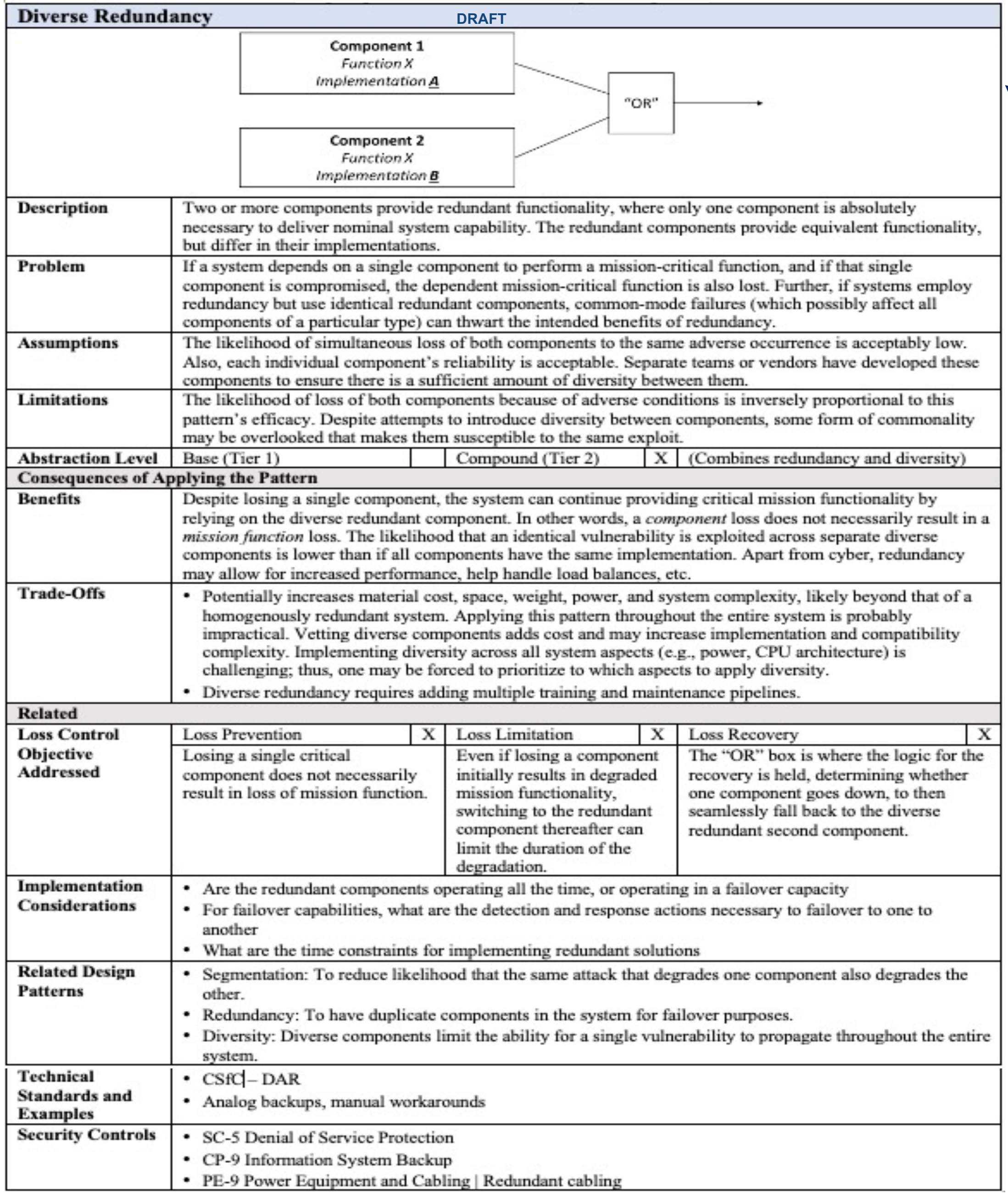
Assumptions

Limitations

Abstraction Level

Consequences of App

Design Pattern Title	DRAFT
Diagram illustrating pattern components in relation to one another	
Description	Summary of the main ideas about the illustrated design pattern.
Problem	An undesirable potential circumstance for which the pattern may provide a mitigating solution.
Assumptions	Conditions that must be true for proper application of the pattern. Assumptions provide context and dependences for the pattern's application.
Limitations	Cautions regarding the pattern's efficacy and applicable contexts.
Abstraction Level	An enumerated pattern category, one of either "base" or "compound." A *base* pattern is the lowest decomposition level. Combining base patterns results in *compound* patterns.
Consequences of Applying the Pattern	
Benefits	Desirable outcomes the pattern may enable; specifically, outcomes that address the stated problem.
Trade-Offs	Acknowledgment of possible consequences imposed by applying the pattern, possibly necessitating some compromises to otherwise beneficial system qualities elsewhere.
Related	
Loss Control Objective Addressed	An enumerated set of loss-related goals (from "Design Tenets Review," Draft, MITRE Corporation). The pattern can support one or more of these goals. The term "loss" may apply both to a component and to a mission capability, as specified in the completed template. The loss is usually in the context of mission capability or other end or outcome. The pattern may enable the system to: • Prevent the loss from occurring • Limit the extent of the loss • Fully or partially recover from the loss
Implementation Considerations	To help bridge the gap between abstract concept and specific implementation, this section provides considerations on how to implement the design pattern.
Related Design Patterns	Additional design patterns that, when used in conjunction with this pattern, contribute to solving this pattern's problem scope. Patterns listed here may complement this pattern to overcome limitations or combine to yield a more powerful capability.
Related Design Principles	This is a placeholder for tracing design patterns to draft MITRE Design Tenets document. Will be added once document is finalized.
Technical Standards and Examples	Texts, standards, applications, and/or examples that present the design pattern and/or describe its employed use cases. The references listed here may call the design pattern by a different name, but the application still meets the spirit and intent of the design pattern described in the template.
Potential Security Controls	The given pattern could be used to satisfy the listed security controls in NIST SP800-53. This is not meant to be a comprehensive list, only a subset of examples.



Diverse Redundancy

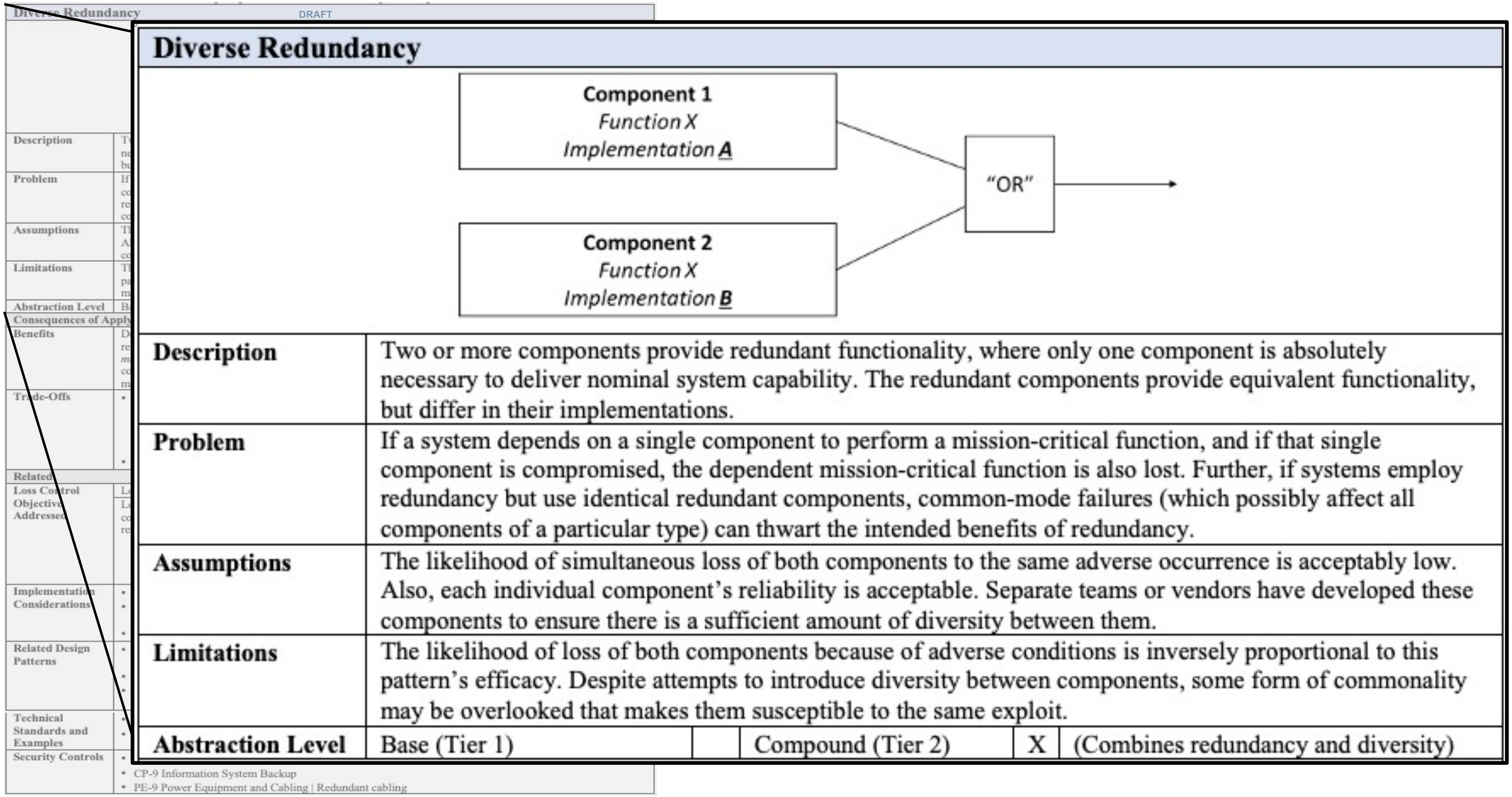
DRAFT

Component 1
Function X
Implementation A
Component 2
Function X
Implementation B
"OR"

Description	Two or more components provide redundant functionality, where only one component is absolutely necessary to deliver nominal system capability. The redundant components provide equivalent functionality, but differ in their implementations.					
Problem	If a system depends on a single component to perform a mission-critical function, and if that single component is compromised, the dependent mission-critical function is also lost. Further, if systems employ redundancy but use identical redundant components, common-mode failures (which possibly affect all components of a particular type) can thwart the intended benefits of redundancy.					
Assumptions	The likelihood of simultaneous loss of both components to the same adverse occurrence is acceptably low. Also, each individual component's reliability is acceptable. Separate teams or vendors have developed these components to ensure there is a sufficient amount of diversity between them.					
Limitations	The likelihood of loss of both components because of adverse conditions is inversely proportional to this pattern's efficacy. Despite attempts to introduce diversity between components, some form of commonality may be overlooked that makes them susceptible to the same exploit.					
Abstraction Level	Base (Tier 1)		Compound (Tier 2)	X	(Combines redundancy and diversity)	
Consequences of Applying the Pattern						
Benefits	Despite losing a single component, the system can continue providing critical mission functionality by relying on the diverse redundant component. In other words, a *component* loss does not necessarily result in a *mission function* loss. The likelihood that an identical vulnerability is exploited across separate diverse components is lower than if all components have the same implementation. Apart from cyber, redundancy may allow for increased performance, help handle load balances, etc.					
Trade-Offs	• Potentially increases material cost, space, weight, power, and system complexity, likely beyond that of a homogenously redundant system. Applying this pattern throughout the entire system is probably impractical. Vetting diverse components adds cost and may increase implementation and compatibility complexity. Implementing diversity across all system aspects (e.g., power, CPU architecture) is challenging; thus, one may be forced to prioritize to which aspects to apply diversity. • Diverse redundancy requires adding multiple training and maintenance pipelines.					
Related						
Loss Control Objective Addressed	Loss Prevention	X	Loss Limitation	X	Loss Recovery	X
	Losing a single critical component does not necessarily result in loss of mission function.		Even if losing a component initially results in degraded mission functionality, switching to the redundant component thereafter can limit the duration of the degradation.		The "OR" box is where the logic for the recovery is held, determining whether one component goes down, to then seamlessly fall back to the diverse redundant second component.	
Implementation Considerations	• Are the redundant components operating all the time, or operating in a failover capacity • For failover capabilities, what are the detection and response actions necessary to failover to one to another • What are the time constraints for implementing redundant solutions					
Related Design Patterns	• Segmentation: To reduce likelihood that the same attack that degrades one component also degrades the other. • Redundancy: To have duplicate components in the system for failover purposes. • Diversity: Diverse components limit the ability for a single vulnerability to propagate throughout the entire system.					
Technical Standards and Examples	• CSfC – DAR • Analog backups, manual workarounds					
Security Controls	• SC-5 Denial of Service Protection • CP-9 Information System Backup • PE-9 Power Equipment and Cabling | Redundant cabling					

Subset of Design Patterns Developed:

- Redundancy
- Diverse Redundancy
- Data Diode
- Segmentation
- Authentication
- Authorization
- Trust Anchor
- Watch Dog
- Data Collection
- Analytics
- Alerts
- Response
- Load from Known State
- …. & More



DRAFT
Component 1
Function X
Implementation ***A***
Component 2
Function X
Implementation ***B***
"OR"

Diverse Redundancy

<table>
<tr><td><b>Description</b></td><td colspan="5">Two or more components provide redundant functionality, where only one component is absolutely necessary to deliver nominal system capability. The redundant components provide equivalent functionality, but differ in their implementations.</td></tr>
<tr><td><b>Problem</b></td><td colspan="5">If a system depends on a single component to perform a mission-critical function, and if that single component is compromised, the dependent mission-critical function is also lost. Further, if systems employ redundancy but use identical redundant components, common-mode failures (which possibly affect all components of a particular type) can thwart the intended benefits of redundancy.</td></tr>
<tr><td><b>Assumptions</b></td><td colspan="5">The likelihood of simultaneous loss of both components to the same adverse occurrence is acceptably low. Also, each individual component's reliability is acceptable. Separate teams or vendors have developed these components to ensure there is a sufficient amount of diversity between them.</td></tr>
<tr><td><b>Limitations</b></td><td colspan="5">The likelihood of loss of both components because of adverse conditions is inversely proportional to this pattern's efficacy. Despite attempts to introduce diversity between components, some form of commonality may be overlooked that makes them susceptible to the same exploit.</td></tr>
<tr><td><b>Abstraction Level</b></td><td>Base (Tier 1)</td><td></td><td>Compound (Tier 2)</td><td>X</td><td>(Combines redundancy and diversity)</td></tr>
</table>

Description

Problem

Assumptions

Limitations

Abstraction Level

Consequences of Apply

Benefits

Trade-Offs

Related

Loss Control Objective Addressed

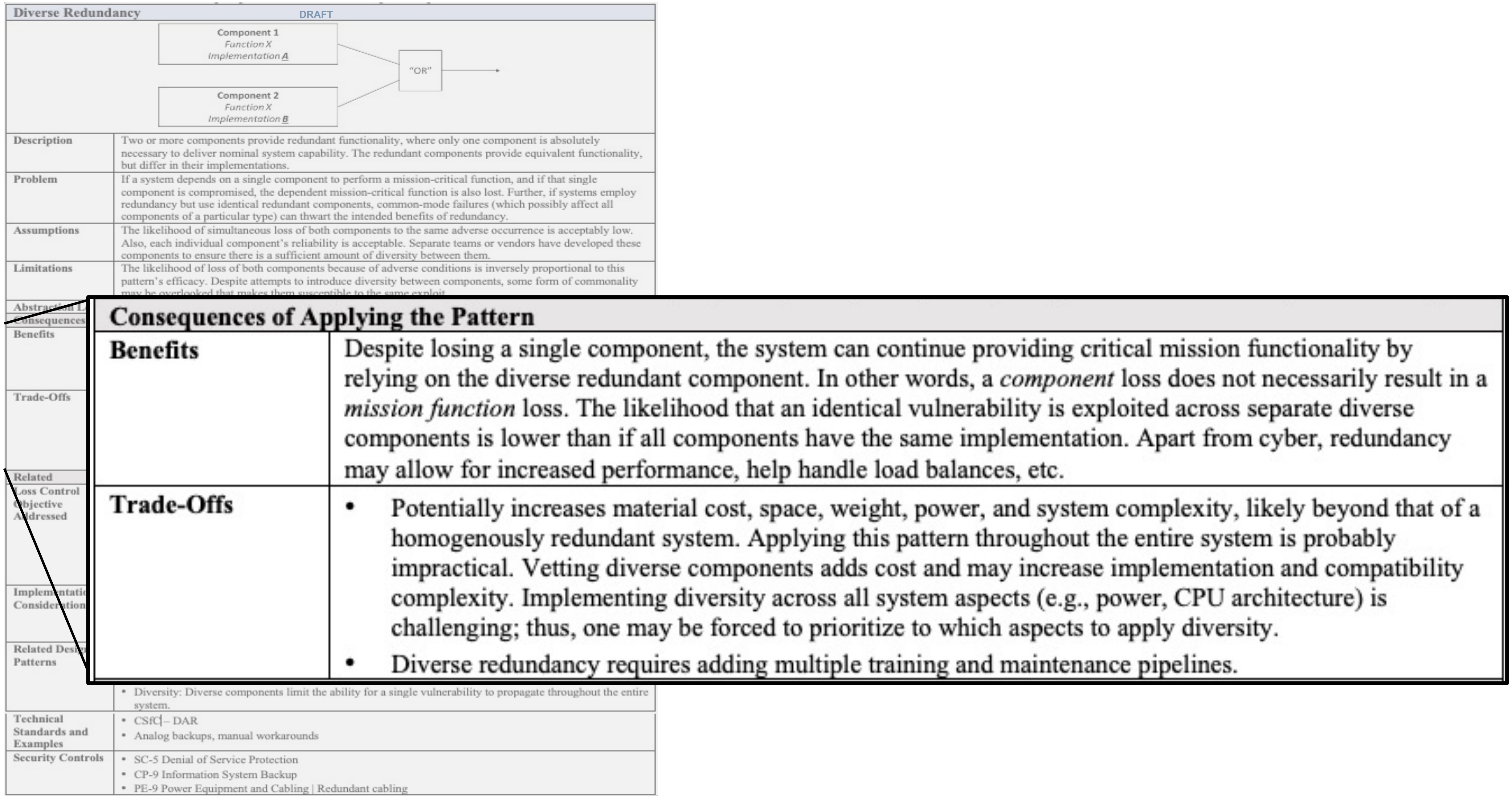
Implementation Considerations

Related Design Patterns

Technical Standards and Examples

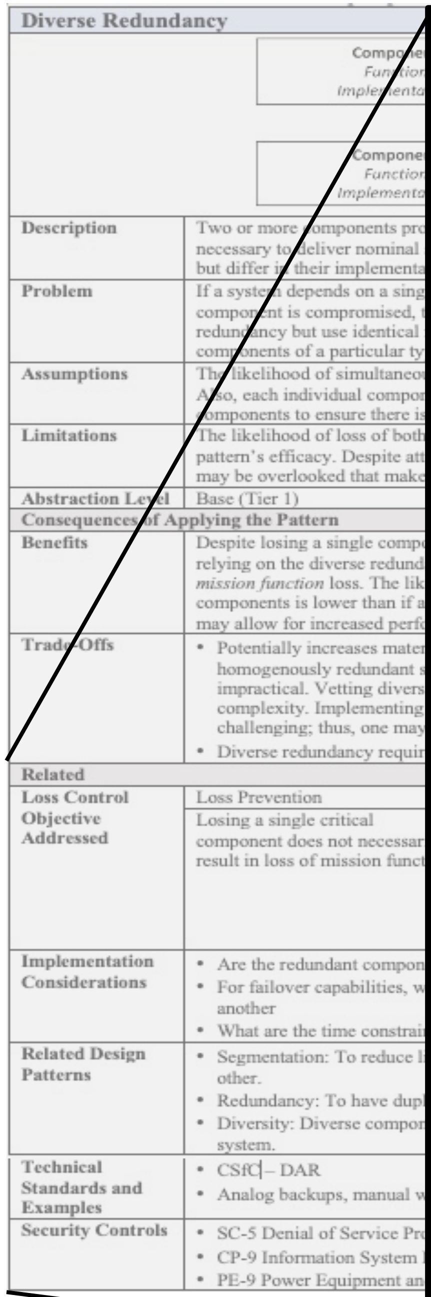
Security Controls

- CP-9 Information System Backup
- PE-9 Power Equipment and Cabling | Redundant cabling

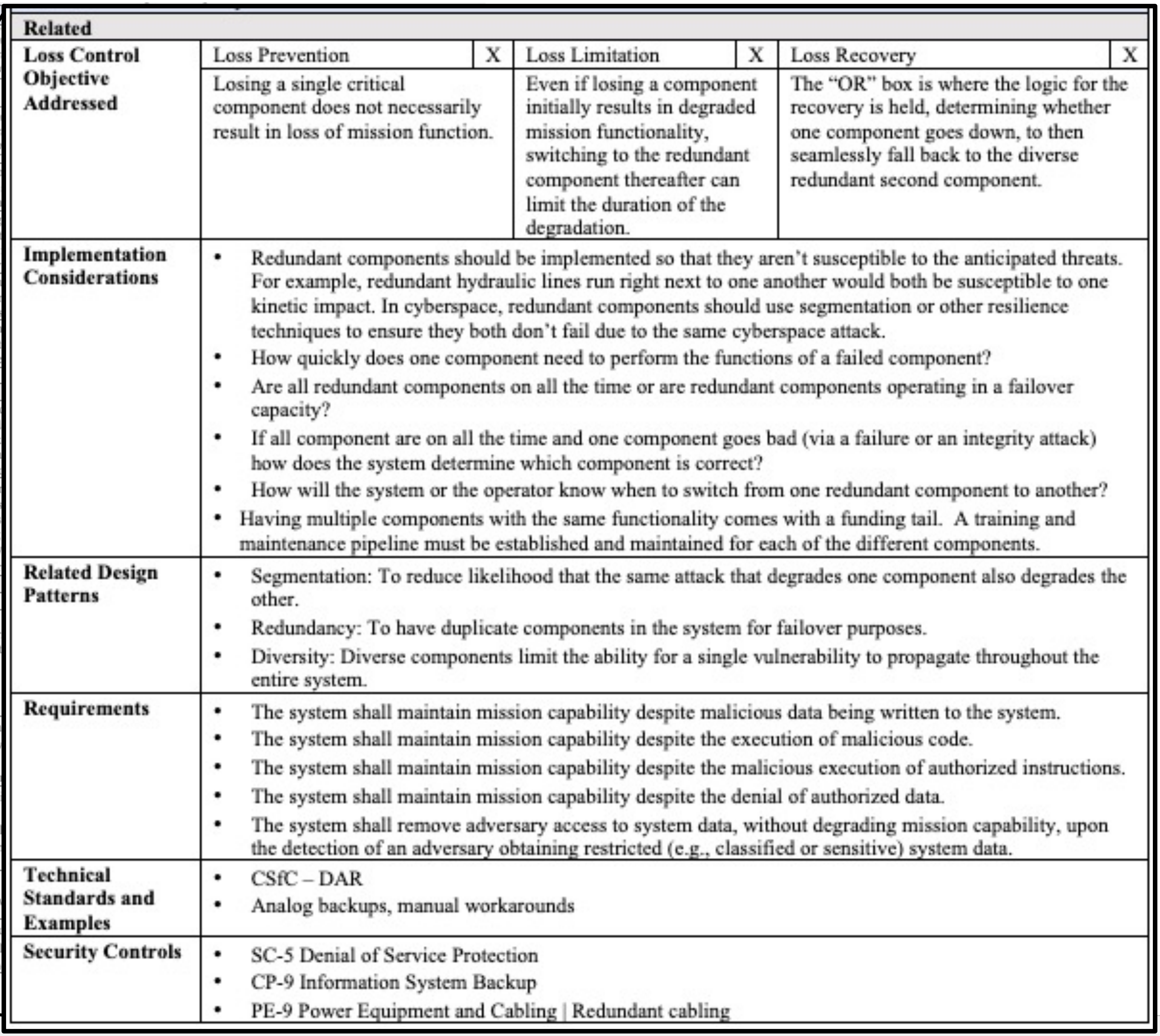

DRAFT
Component 1
Function X
Implementation A
Component 2
Function X
Implementation B
"OR"

Diverse Redundancy	
Description	Two or more components provide redundant functionality, where only one component is absolutely necessary to deliver nominal system capability. The redundant components provide equivalent functionality, but differ in their implementations.
Problem	If a system depends on a single component to perform a mission-critical function, and if that single component is compromised, the dependent mission-critical function is also lost. Further, if systems employ redundancy but use identical redundant components, common-mode failures (which possibly affect all components of a particular type) can thwart the intended benefits of redundancy.
Assumptions	The likelihood of simultaneous loss of both components to the same adverse occurrence is acceptably low. Also, each individual component's reliability is acceptable. Separate teams or vendors have developed these components to ensure there is a sufficient amount of diversity between them.
Limitations	The likelihood of loss of both components because of adverse conditions is inversely proportional to this pattern's efficacy. Despite attempts to introduce diversity between components, some form of commonality may be overlooked that makes them susceptible to the same exploit.
Abstraction L	
Consequences	
Benefits	
Trade-Offs	
Related Loss Control Objective Addressed	
Implementati Consideration	
Related Desi Patterns	- Diversity: Diverse components limit the ability for a single vulnerability to propagate throughout the entire system.
Technical Standards and Examples	- CSfC– DAR - Analog backups, manual workarounds
Security Controls	- SC-5 Denial of Service Protection - CP-9 Information System Backup - PE-9 Power Equipment and Cabling | Redundant cabling

Consequences of Applying the Pattern	
Benefits	Despite losing a single component, the system can continue providing critical mission functionality by relying on the diverse redundant component. In other words, a *component* loss does not necessarily result in a *mission function* loss. The likelihood that an identical vulnerability is exploited across separate diverse components is lower than if all components have the same implementation. Apart from cyber, redundancy may allow for increased performance, help handle load balances, etc.
Trade-Offs	- Potentially increases material cost, space, weight, power, and system complexity, likely beyond that of a homogenously redundant system. Applying this pattern throughout the entire system is probably impractical. Vetting diverse components adds cost and may increase implementation and compatibility complexity. Implementing diversity across all system aspects (e.g., power, CPU architecture) is challenging; thus, one may be forced to prioritize to which aspects to apply diversity. - Diverse redundancy requires adding multiple training and maintenance pipelines.



Diverse Redundancy	
	Compone Function Implementa Compone Function Implementa
Description	Two or more components pro necessary to deliver nominal but differ in their implementa
Problem	If a system depends on a sing component is compromised, t redundancy but use identical components of a particular ty
Assumptions	The likelihood of simultaneo Also, each individual compon components to ensure there is
Limitations	The likelihood of loss of both pattern's efficacy. Despite att may be overlooked that make
Abstraction Level	Base (Tier 1)
Consequences of Applying the Pattern	
Benefits	Despite losing a single compo relying on the diverse redund *mission function* loss. The lik components is lower than if a may allow for increased perfo
Trade-Offs	• Potentially increases mater homogenously redundant s impractical. Vetting divers complexity. Implementing challenging; thus, one may • Diverse redundancy requir
Related	
Loss Control Objective Addressed	Loss Prevention Losing a single critical component does not necessar result in loss of mission funct
Implementation Considerations	• Are the redundant compon • For failover capabilities, w another • What are the time constrai
Related Design Patterns	• Segmentation: To reduce li other. • Redundancy: To have dupl • Diversity: Diverse compon system.
Technical Standards and Examples	• CSfC – DAR • Analog backups, manual w
Security Controls	• SC-5 Denial of Service Pro • CP-9 Information System I • PE-9 Power Equipment an



Related						
Loss Control Objective Addressed	Loss Prevention	X	Loss Limitation	X	Loss Recovery	X
	Losing a single critical component does not necessarily result in loss of mission function.		Even if losing a component initially results in degraded mission functionality, switching to the redundant component thereafter can limit the duration of the degradation.		The "OR" box is where the logic for the recovery is held, determining whether one component goes down, to then seamlessly fall back to the diverse redundant second component.	
Implementation Considerations	• Redundant components should be implemented so that they aren't susceptible to the anticipated threats. For example, redundant hydraulic lines run right next to one another would both be susceptible to one kinetic impact. In cyberspace, redundant components should use segmentation or other resilience techniques to ensure they both don't fail due to the same cyberspace attack. • How quickly does one component need to perform the functions of a failed component? • Are all redundant components on all the time or are redundant components operating in a failover capacity? • If all component are on all the time and one component goes bad (via a failure or an integrity attack) how does the system determine which component is correct? • How will the system or the operator know when to switch from one redundant component to another? • Having multiple components with the same functionality comes with a funding tail. A training and maintenance pipeline must be established and maintained for each of the different components.					
Related Design Patterns	• Segmentation: To reduce likelihood that the same attack that degrades one component also degrades the other. • Redundancy: To have duplicate components in the system for failover purposes. • Diversity: Diverse components limit the ability for a single vulnerability to propagate throughout the entire system.					
Requirements	• The system shall maintain mission capability despite malicious data being written to the system. • The system shall maintain mission capability despite the execution of malicious code. • The system shall maintain mission capability despite the malicious execution of authorized instructions. • The system shall maintain mission capability despite the denial of authorized data. • The system shall remove adversary access to system data, without degrading mission capability, upon the detection of an adversary obtaining restricted (e.g., classified or sensitive) system data.					
Technical Standards and Examples	• CSfC – DAR • Analog backups, manual workarounds					
Security Controls	• SC-5 Denial of Service Protection • CP-9 Information System Backup • PE-9 Power Equipment and Cabling | Redundant cabling					

Next Steps

- Integrate design pattern into CRWS-BoK repository

- Demonstrate design pattern applicability and interoperability

- Continue development and refinement of existing patterns and template

JOHNS HOPKINS
APPLIED PHYSICS LABORATORY