

Game Aggregator Disaster Recovery Plan Template

A comprehensive disaster recovery plan (DRP) for a game aggregator should include detailed strategies across four key phases: **Preparedness/Mitigation, Response, Recovery, and Testing/Maintenance.**

Key ingredients:

1. Plan Overview and Management Summary

- **Revision History:** A log of all updates to the DRP.
- **Purpose and Scope:** A high-level description of what the plan covers: the aggregation platform, primary databases, API integrations, and network infrastructure.
- **Business Continuity Objectives:** Primary goals include minimal data loss (near-zero RPO for transactional data) and rapid system restoration (strict RTOs, potentially minutes of downtime).
- **Key Personnel & Management Contacts:** An up-to-date call tree with contact details for the Disaster Recovery Team (DRT), management, and alternates.

2. Business Impact Analysis (BIA) and Risk Assessment

- **Critical Systems Identification:**
 - Aggregation Platform/Game Hub
 - Transaction Databases (bets, wins, player data)
 - API Gateways for operator and developer connectivity
 - Monitoring and Reporting Tools
 - Billing and Financial Systems
- **Regulatory Compliance Analysis:** Identification of all regulatory requirements that mandate specific RTOs, data integrity, and reporting obligations during an outage.
- **Threat Matrix:** Assessment of threats specific to a high-transaction environment, including DDoS attacks, system failure, ransomware, and network outages.

3. Incident Response Plan

- **Activation Criteria:** Clear criteria for declaring a disaster (e.g., "Main API gateway offline for 5 minutes").
- **Roles and Responsibilities:**
 - **Incident Commander:** Overall authority for the response and recovery.
 - **Network Team:** Responsible for restoring connectivity and managing failover.
 - **Database Team:** Responsible for data restoration and integrity checks.
 - **Compliance & Legal Officer:** Ensures all recovery actions adhere to regulatory obligations.
- **Incident Triage Procedures:** Steps to immediately stabilize the environment, prevent further data loss, and redirect traffic to alternate systems.

4. Data Backup, Redundancy, and Recovery Strategy

- **Automated Backups:** Implement automated, frequent (potentially continuous) backups of all transactional data, configurations, and system logs.
- **Off-site/Cloud Storage:** Encrypted backups stored in secure, multi-region cloud environments to ensure geographical separation from the primary data center.
- **Redundancy and Failover Mechanisms:**
 - **Hot Site/Active-Active Architecture:** Maintain a fully operational, redundant secondary environment (hot site) that can take over operations instantly in case of primary site failure.
 - **Automated Failover:** Use load balancers and DNS routing to automatically switch traffic to the alternate site with minimal manual intervention.
- **Data Integrity Verification:** Procedures for ensuring that all data is consistent across primary and backup sites and meets compliance standards (RNG certification data integrity).

5. Infrastructure and Vendor Coordination

- **Infrastructure Inventory:** An up-to-date list of all network equipment, servers, software licenses, and third-party tools.

- **Vendor Contact & SLA Review:** A list of all game developers (suppliers) and operators (customers) with key contacts. The DRP must align with agreed-upon Service Level Agreements (SLAs) regarding uptime and notification times including **escalation paths** to ensure their support during a crisis.
- **Alternate Facilities:** While operations may be entirely cloud-based, a physical location for core DRT assembly (e.g., a secondary office or pre-arranged meeting spot) should be defined.

6. Communication and Reporting Plan

- **Internal Communication:** Use redundant channels (e.g., satellite phones, encrypted messaging apps, status page) to communicate with the DRT and employees.
- **External Communication:**
 - Pre-drafted templates for notifying operators, game developers, and regulatory bodies.
 - Protocols for providing regular, transparent updates to all stakeholders to maintain trust and manage expectations.
- **Regulatory Reporting:** A process for officially reporting the incident and recovery status to all relevant gaming authorities within specified timeframes.

7. Testing, Training, and Maintenance

- **Regular Testing (Simulations):** Conduct frequent, mandatory DRP testing (quarterly or semi-annually). These tests should include full-scale failover simulations involving all internal teams and potentially key external partners.
- **Training:** Provide ongoing training for all relevant staff on their roles and responsibilities during a disaster.
- **Post-Incident Review:** A formal process for conducting a root cause analysis (RCA) after any test or actual incident to identify weaknesses and continuously improve the plan.
- **Plan Review Cycle:** A commitment to review and update the DRP at least annually or after any significant change to the platform's architecture or regulatory environment

Game Developer Disaster Recovery Plan Template

A Disaster Recovery Plan (DRP) for a game developer is designed to protect intellectual property, ensure the continuity of the development pipeline, and minimize project delays.

Essential components:

1. Plan Overview and Documentation

- **Revision History:** A log of all changes to the DRP.
- **Purpose and Scope:** A high-level description of what systems the plan covers (e.g., source code, asset servers, build pipeline) and the scenarios it addresses (e.g., data corruption, ransomware, facility loss).
- **Key Personnel Contact List:** An up-to-date list of all team members, their roles during a disaster, and multiple contact methods (personal phone, email, emergency messaging app).

2. Business Impact Analysis (BIA) and Risk Assessment

- **Critical Assets Identification:**
 - Source Code Repositories (Git, Perforce, SVN)
 - Game Engine (Unity, Unreal, custom)
 - Digital Art/Sound Assets
 - Build Servers & Environments
 - Project Management & Communication Tools (Jira, Slack)
 - Testing/QA Environments
- **Recovery Objectives (RTO/RPO):**
 - **RTO (Recovery Time Objective):** Maximum acceptable downtime for each critical asset (e.g., 4 hours for source code repository).
 - **RPO (Recovery Point Objective):** Maximum acceptable data loss for each asset (e.g., 15 minutes of work for source code).
- **Threat Matrix:** A list of potential threats (e.g., ransomware, hardware failure, fire) and their potential impact on development timelines and IP.

3. Incident Response Plan

- **Activation Criteria:** Clear criteria for when the DRP is activated (e.g., "Main VCS server is offline for more than 30 minutes").
- **Roles and Responsibilities:**
 - **Incident Manager:** Overall command and decision-making authority.
 - **Technical Team Leads:** Responsible for recovery of specific systems (e.g., Engineering Lead for code, Art Director for assets).
 - **Communications Officer:** Responsible for internal/external communication.
- **Incident Triage Procedures:** Steps to assess the situation, contain the damage, and classify the severity of the disaster.

4. Data Backup and Restoration Strategy

- **Backup Schedule and Methods:**
 - Automated backups of all source code, assets, and databases.
 - Frequency defined by RPO (e.g., continuous synchronization for code, daily for assets).
- **Storage Locations:** Encrypted backups stored off-site or in geographically diverse cloud regions.
- **Restoration Procedures (Runbooks):** Step-by-step guides for restoring each critical system:
 - *Example:* "Procedure for restoring Perforce server from Azure cloud backup."
- **Backup Verification:** A protocol for regularly testing that backups are restorable and functional.

5. Infrastructure and Continuity (Remote Work Focus)

- **Alternate Work Environment:** A plan for all employees to transition to remote work seamlessly.
- **Hardware and Software Provisioning:** Procedures for deploying pre-configured laptops or accessing secure virtual desktop environments if primary hardware is lost or damaged.

- **Vendor Coordination:** A list of critical vendors (e.g., engine providers like Unity or Unreal Engine, cloud providers, hardware suppliers) with contact information and support agreements (SLAs).

6. Communication Plan

- **Internal Communication Channels:** Use redundant communication tools (e.g., SMS, a dedicated Slack/Discord channel, or a status page) that do not rely on the potentially compromised primary network.
- **External Communication:**
 - Protocols for informing publishers, investors, and platform partners (e.g., Steam, Epic Games Store, console partners) about potential project delays or security breaches.
 - A pre-approved statement template for public relations purposes.

7. Testing, Training, and Maintenance

- **Testing Schedule:** Plan for annual or semi-annual DRP testing (e.g., a "code restoration drill" or a "simulated office outage").
- **Training:** Provide ongoing training for all staff on DRP roles and general data hygiene practices.
- **Review Cycle:** A commitment to review and update the DRP whenever a major project milestone is hit, new technology is adopted, or a significant change in the business environment occurs.