

DELIVER!™

The tactical game of service management and DevOps

by Nigel Hopkins

“DELIVER!” is a tactical business simulation game set in the challenging world of IT service delivery and improvement. It focuses on creating the best tactical approach to deliver the most benefit to your customers whilst continually improving the resilience of your IT landscape. The gameplay progresses in a series of rounds with each round seeing input from the service desk that sets the operational challenges for the round.

The game develops a cross functional understanding of the holistic nature of incident, problem, and change areas of ITIL. Playing the game, provides the basis to establish a practical tactical approach to prioritisation of activities.

The interactive nature of the game develops many soft skills including communications, teamwork, time management, and leadership.

In the game, each team is managing their own service along with its underlying systems and technical platforms. They are also supporting, from an operational point, other shared services, leading to a semi-cooperative approach in the game-play. The players must continually improve their services whilst maintaining their support promise to their customers.

All of the operational activities put a heavy demand on the limited resources available to each team, and it is this aspect that leads to the development of a successful tactical approach.

OBJECTIVE

You are the lead in one of a number of centrally located teams, with end-to-end service management responsibilities. Each team is responsible for their own solution and have shared responsibility for several other solutions.

You are tasked with improving your service's functionality and, at the same time, ensuring that any incidents are dealt with effectively, preferably without breaching your support promise.

Whilst you have your own service to take care of, you, along with the other teams, also have a shared responsibility for some other services. You gain influence when helping others, either the

shared services or other teams. However, you lose influence if you fail to keep your support promise.

You will gain business value by improving your service, especially where the improvements align with the needs of your customer. In the long term, creating resilience across all services is of value to the business as well and will be rewarded accordingly

In the end it all comes down to the player that manages to create the most business value.

Page 1-6

CONCEPTS and CONTENTS
Boards, cards and elements

Page 7-8

SETTING UP

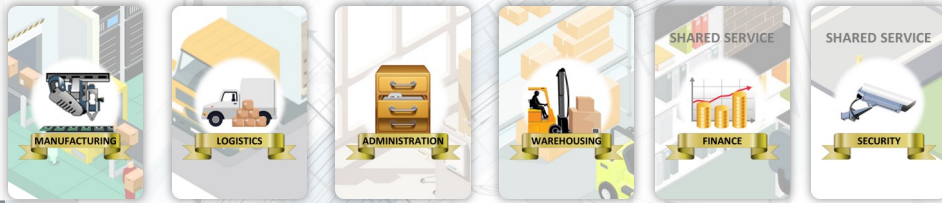
Page 9-12

SEQUENCE OF PLAY

Page 13-16

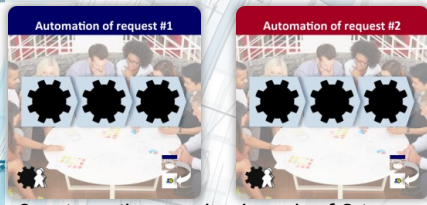
ACTIONS

WHAT'S IN THE BOX?



6 operations decks, each of 16 event cards

Centre board



8 automation cards, 4 each of 2 types



Customer improvement request deck, 16 cards

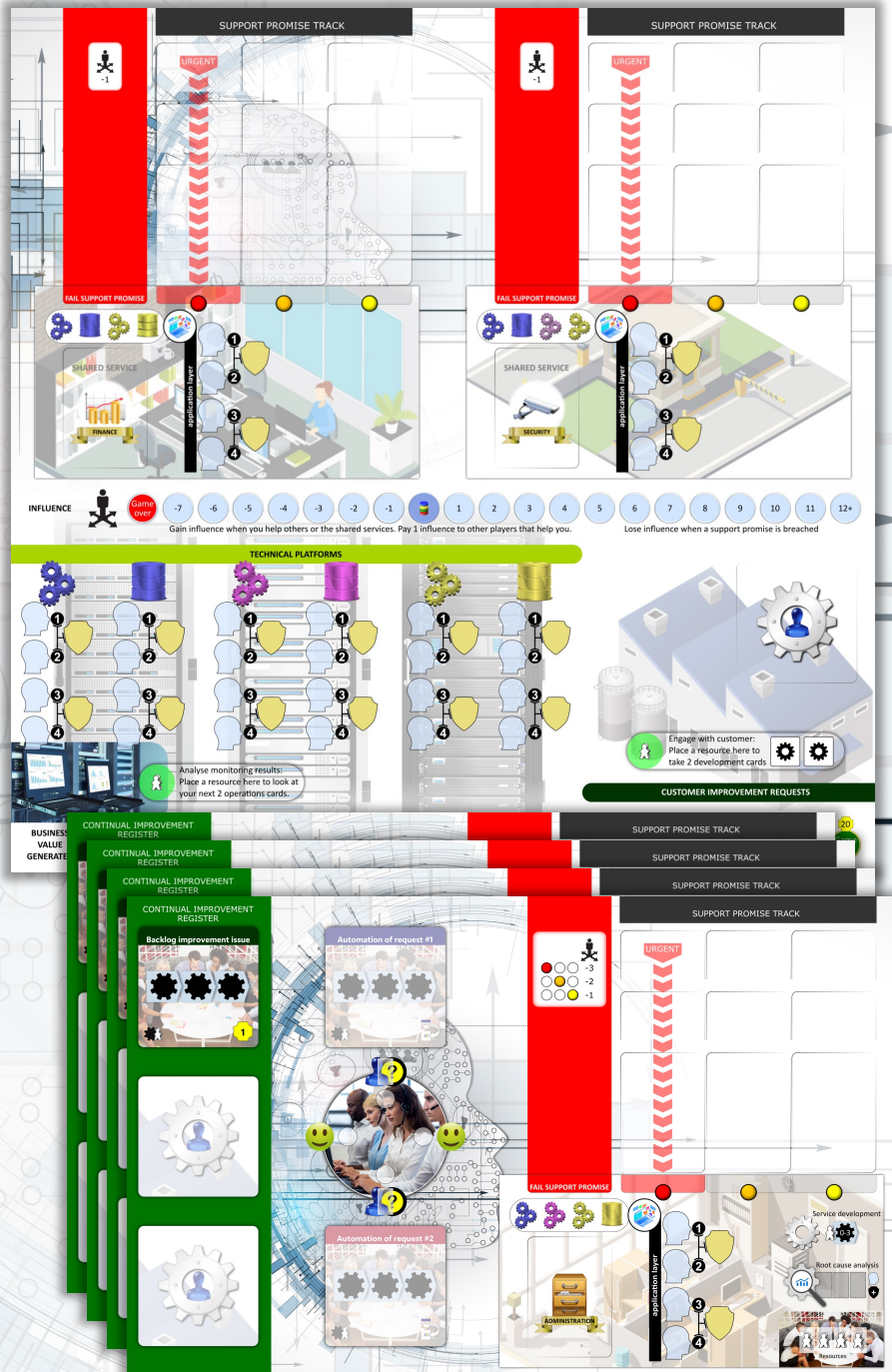
1 service desk marker in each of 4 colours (blue, green, red, and yellow)



16 player discs in each of 4 colours (blue, green, red, and yellow)



4 resources in each of 4 colours (blue, green, red, and yellow)



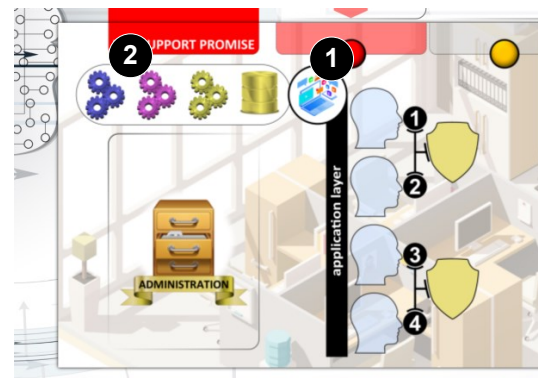
4 service management mats

Key concepts

The services

There are six services represented in the game; Administration, Finance, Logistics, Manufacturing, Security, and Warehousing. Each of the services has a similar approach with an application layer (1) and an infrastructure layer comprising of four technical platforms (2), from the six available (see below). To the right is the administration service, with its application layer and 4 supporting technical platforms, as an example.

The application layer has four vulnerabilities, each of which is being monitored. Changes in the status of the monitored vulnerabilities creates events which are assessed against defined thresholds. If a threshold is breached, an exception

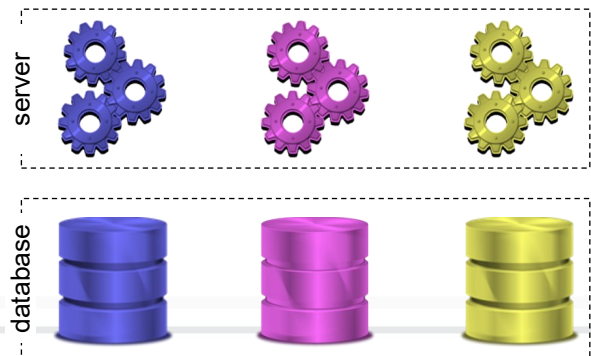


Technical platforms

The architecture behind DELIVER! is one of six platforms in three platform areas. Each area (see right) is designated by a different colour; Blue, Purple, or Yellow.

Each platform area comprises two platforms, one for the operating servers, shown to the right along the top row, and one for databases, typically database, shown along the bottom row.

As with the application layer, each technical platform has four vulnerabilities, each of which is being monitored. Changes in the status of the monitored vulnerabilities creates events which are assessed against defined thresholds. If a threshold is breached, an exception event is generated with an error of type 1-4.



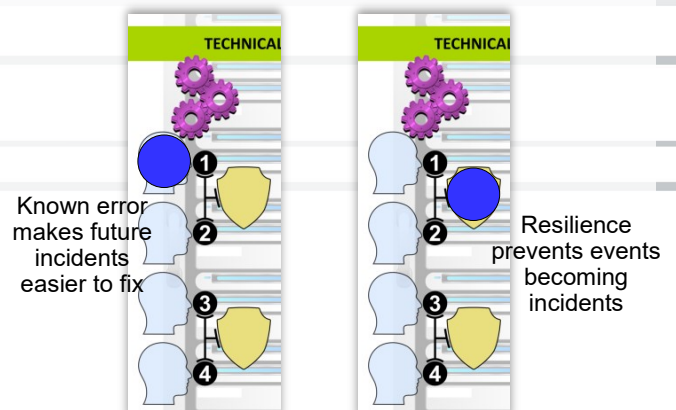
Addressing vulnerabilities and creating resilience

Initially, when an event is generated it will normally be considered to be an exception event and lead to the creation of an incident. The incident will be on the same infrastructure platform or application layer as the event. It will also have the same mode of failure as the error type shown on the event.

To avoid an incident or to lessen its effect there are two options available.

Firstly, if an incident is experienced and resolved, the knowledge gained can be documented as a known error in a knowledge article. In future, this will take less resources to resolve.

By creating resilience, an event is downgraded to an informational event and does not become an incident. Resilience is created by performing a root cause analysis where a known error has been documented in a knowledge article.



Service desk

The service desk provides the primary interface between the user and the IT organisation for reporting service requests and incidents.

In the game, it is the service requests that are in focus at the service desk, with incidents being automatically

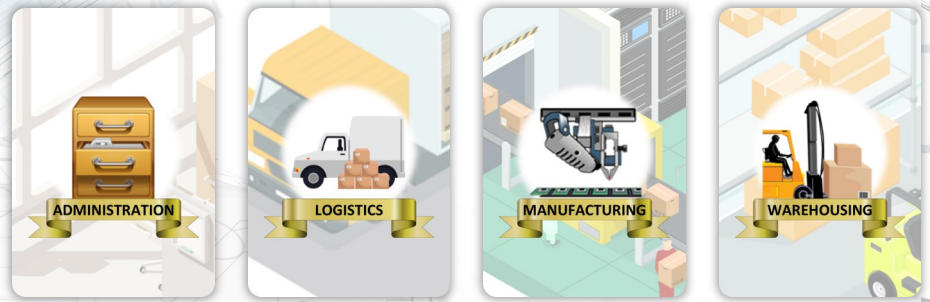
created through exception events.

Service requests are pre-defined support activities that are not incidents though they do require attention to deliver the required outcome to the user.

Event cards and Operations decks

Operations deck - player service

Each managed service has its own operations deck distinguished by its name on the back of the cards. The decks consist of event cards



Top left of the card face is the area in which the event has been triggered being either the application layer, as seen here, or one of the technical platforms supporting the service.

Event cards

Top right of the card is the error type, a number from 1 to 4. During play, this may become a known error or be permanently fixed.

Priority 1 (P1)



Priority 2 (P2)

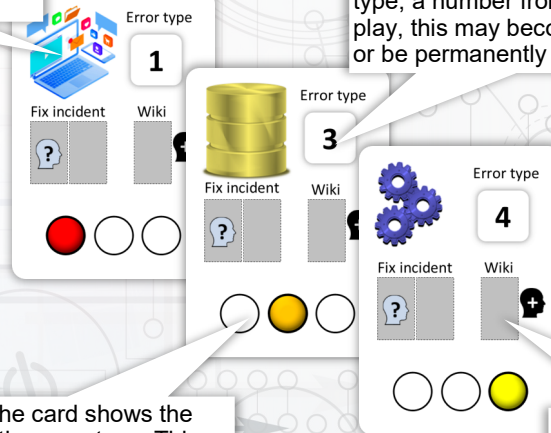


Priority 3 (P3)



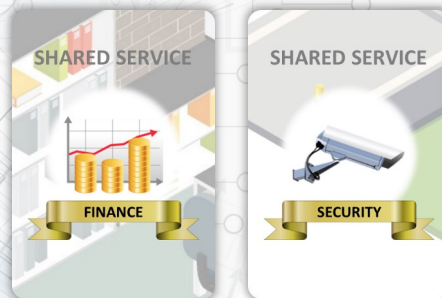
The bottom of the card shows the priority level of the error type. This determines here the card will be placed on the support promise track.

The centre of the card has the area for assigning resources needed to fix the incident and, optionally, create a known error knowledge article.



Operations deck - shared service

Each shared service has its own operations deck distinguished by its name on the back of the cards. When playing solo, or with 3 players/teams, some of the events are already fixed as indicated by an icon just above the priority level. During the game, when the condition is met, in terms of the number of players/teams, the card is immediately discarded



"Already fixed" in a solo game

"Already fixed" in a 3 player game

"Already fixed" in a solo or 3 player game

Request automation cards

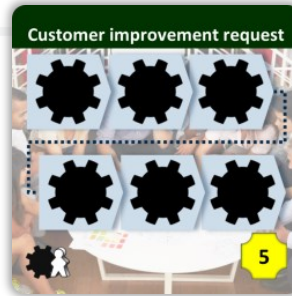
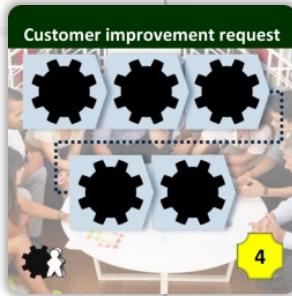
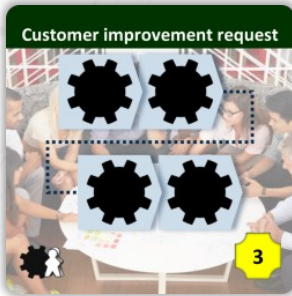
Each player has two automation cards for service request #1 and #2. One side shows the development

steps to automate the request handling and the other side shows that the automation has been completed.



Customer improvement requests

There are a number of Customer improvement request cards detailing service improvement opportunities.



Boards and Mats

Service management mats

The service management mats provide the control area for each player, with Development to the left and Operations to the right. It helps track the main activities and houses the player's managed service mat.

The leftmost section contains the continual improvement register, where development activities are tracked.

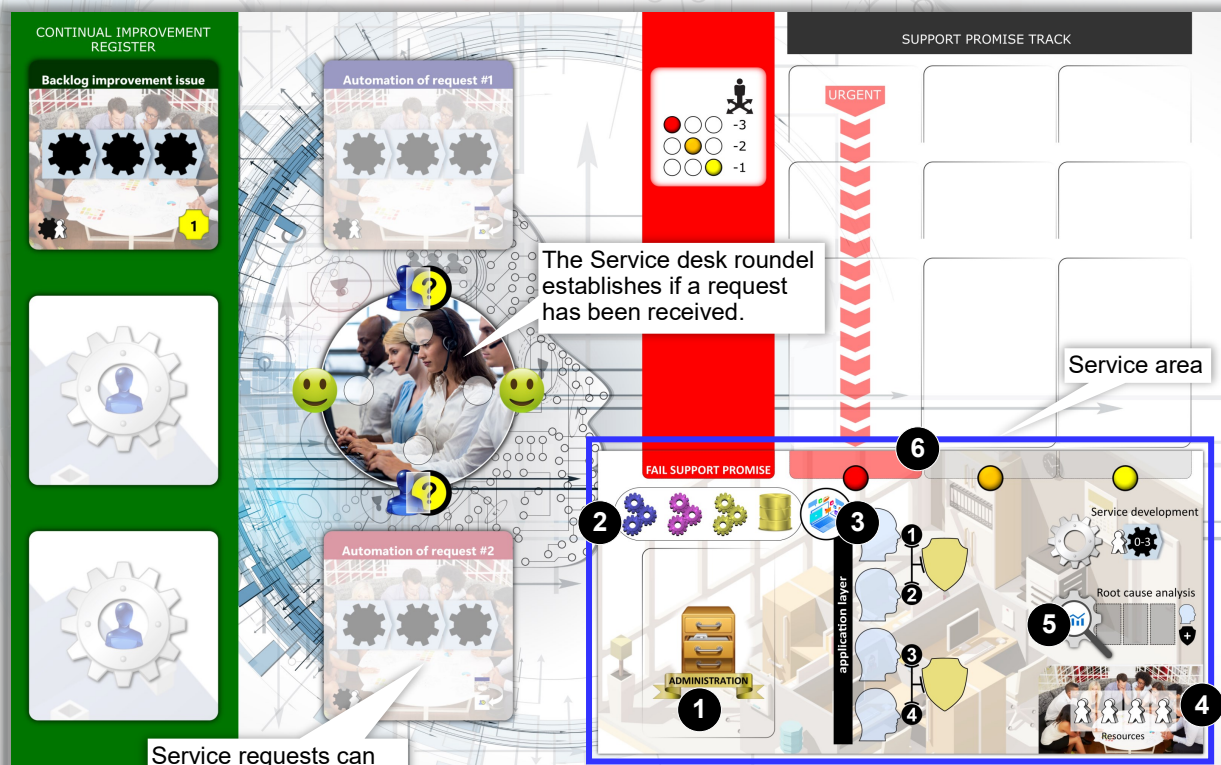
To the right of this section is the service desk roundel and service request automation areas. The roundel is used to establish whether a service request has been received and if so, which one. The two areas for service request automation allow for an automated response to the service requests to be developed and implemented. Without automation, service requests take resources to process.

The right half of the board is the section for managing the service support and DevOps team. At the top is the support promise track, used to keep track of incidents. At the bottom right is the service area for the specific service being managed, in this example it is Administration.

To the left of the service area is the position for the operations deck (1) and the information panel showing the technical platforms that support the service (2). The centre of the service area has the application layer with its four modes of failure, each having a spot for known error and resilience (3).

To the right are the DevOps resources (4) as well as the assignment area for Root cause analysis (5).

Along the top of the service area are the 3 priority sections (6).



Centre board

The centre board contains the Technical platforms (1), the Customer improvement requests (2), and tracks for both Influence (3) and Business value (4). The upper section of the board contains the shared services, jointly supported by all players.

Each technical platform has 4 modes of failure arranged vertically below the platform icon. Against each of these failure modes is a spot for known error (a) and resilience (b).

SETTING UP

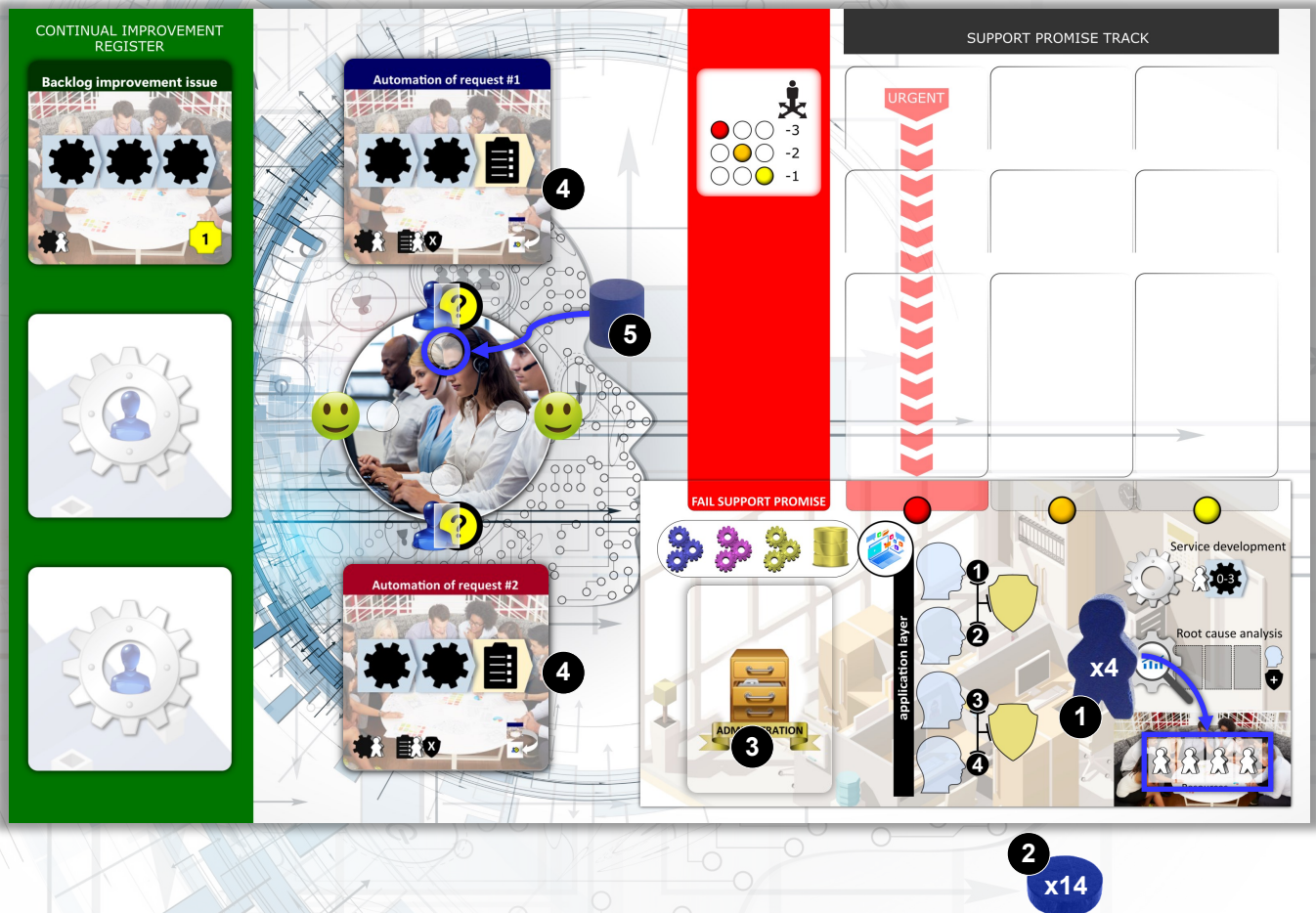
Player managed service setup

Each player:

Place the Service management mat of your chosen service in front of you and, in your chosen colour, take the service desk marker, 4 resources, and 16 player discs.

- 1 Place the resources (wooden figures) into your DevOps resources area.
- 2 Place 14 discs next to your board to form your supply (the remaining 2 will be used later).

- 3 Shuffle your service's Operations deck and place it face-down in your service area.
- 4 Place one Request automation card of each type onto your service management mat with the development track facing up as shown. Automation of request #1 is placed above the service desk roundel with #2 below.
- 5 Place your Service desk marker in the top position of the service desk roundel.



Centre board setup

The centre board should be placed in reach of all players.

Shared services

The shared services are represented in the upper section of the centre board.

In a 1-2 player game you will not use the Security shared service. Shuffle each shared service operations deck and place onto the corresponding service face down (A).

SUPPORT PROMISE TRACK

URGENT

FAIL SUPPORT PROMISE

ED SERVICE FINANCE

SHARED SECURITY

INFLUENCE

Game over

-7 -6 -5 -4 -3 -2 -1 1 2 3 4 5 6 7 8 9 10 11 12+

Gain influence when you help others or the shared services. Lose influence when a support promise is breached.

TECHNICAL PLATFORMS

Analyse monitoring results: Place a resource here to look at your next 2 operations cards.

CUSTOMER IMPROVEMENT REQUESTS

Engage with customer: Place a resource here to take 2 development cards.

BUSINESS VALUE GENERATED

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

Game over

1 Shuffle the Customer improvement request cards and place them face down on the centre board as shown.

2 Each player places one of their 2 remaining discs on the Influence track and the other on the Business value track at the identified positions.

SEQUENCE OF PLAY

The game can be played semi-cooperatively or fully cooperatively. There will always be some need to cooperate with regards to the shared services but when playing fully cooperatively, the end of game scoring reflects that the goal is to increase the value for the business both in terms of direct business value from improvements, and

indirect business value in terms of increased resilience and reliability of the IT estate. Decide which way you would prefer to play before starting the game.

Introduction

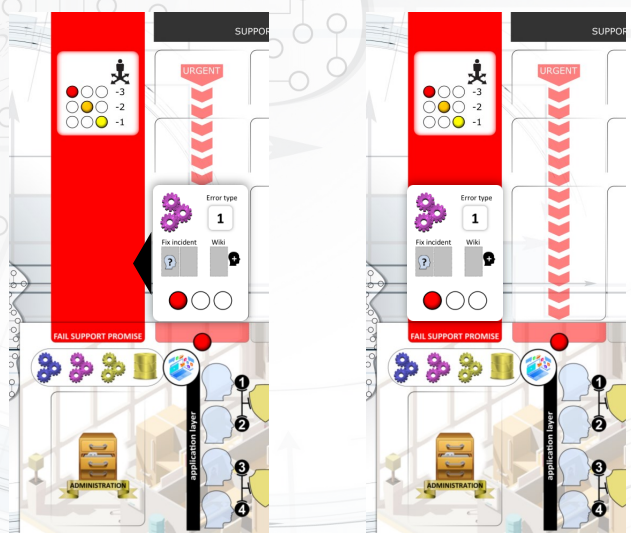
Each round is played through four phases; ADVANCE, REGISTER, ASSIGN, PERFORM. The number of rounds is controlled by the number of cards in each player's service desk and operations deck. Each deck is

normally set to contain 16 cards, giving 16 rounds, though this number can be reduced to have a shorter game. It is recommended that the number of rounds is a minimum of 10.



ADVANCE incidents

As time advances through the game, incidents become more urgent and should be prioritised to avoid breaching the support promise. Move all incidents on the support promise tracks one step to the left. In the example, right, the incident has breached the support promise.



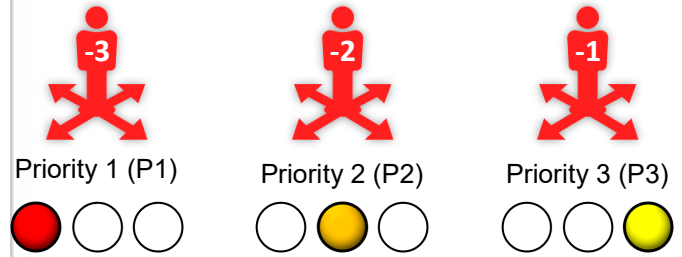
Lose influence if a support promise is breached

Any incidents that are moved into, or remain in the FAIL SUPPORT PROMISE section of your service management mat, or the shared services on the centre board, will incur a loss of influence. Incidents remain until they are fixed.

On your service, the amount of influence lost depends on the original priority of the incident. For the shared services, all players lose 1 influence per incident in the FAIL SUPPORT PROMISE section.

If any player falls below -7 on the influence track the game ends at the end of this phase. Refer to END OF GAME on page 12.

Influence is tracked on the centre board.

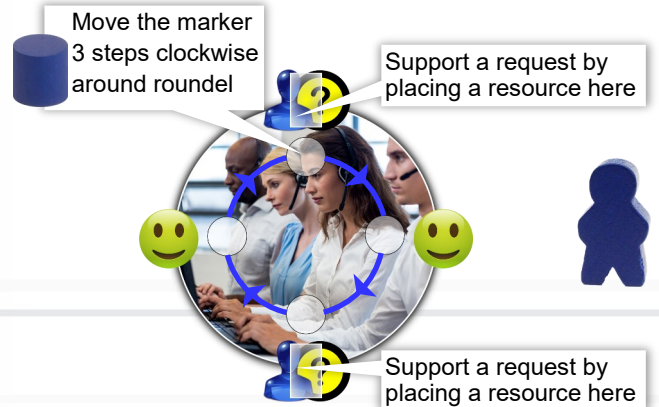


REGISTER service requests and incidents

Each player flips the top card of their operations deck and, moves the Service desk marker clockwise around the service desk roundel, the number of steps shown as the error type. In this example it would be 3 steps



If the marker indicates a service request (top or bottom of the roundel) and you have not automated the request in that position, place one of your resources onto the roundel to deal with the request.



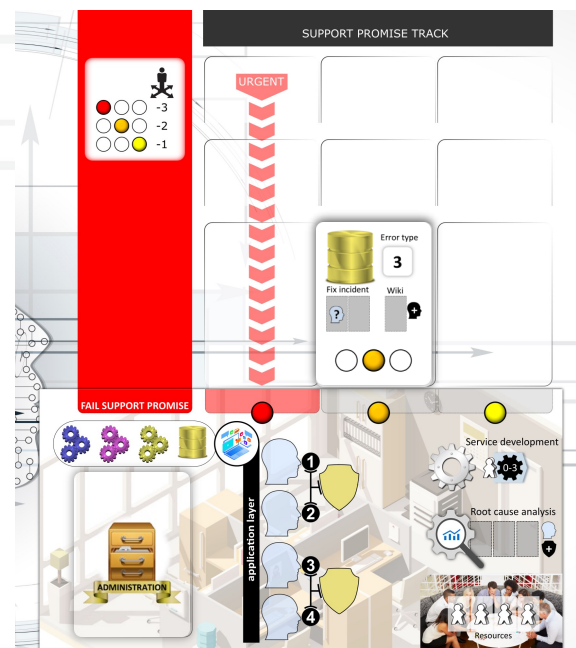
Register incidents

For the revealed event card, check the area and error type of the exception event (see page 3 for more information on the event cards). If resilience has been created, the event is informational only, and does not generate an incident. In this case, the card is discarded.

Without resilience, the event is an exception and will generate an incident which is registered by placing the card onto your support promise track above your service area, matching the priority on the card with the position indicated at the top of your service area. Priority 1 incidents will be placed in the left column of the support promise track, priority 2 incidents will be placed in the middle column, and priority 3 incidents will be placed in the right column.

Finally, reveal the top card from the operations decks of the shared services and place them, to their support promise tracks. Only one card is revealed per shared service. Any "Already fixed" card, or one where resilience has been created, is discarded and no new card is taken this round.

Note, if playing solo or with 3 players/teams, a shared service event may have been fixed - see page 3.



ASSIGN resources

It is now time to assign your available resources by placing them on the various incidents or other action spots.

Assignment is done concurrently for all players but some discussions will be necessary to determine who is doing what, with regards to the shared services. Also some players may want resource assistance from others.

There are 6 different actions to which you can assign your resources:

- Fix incident
- Document known error
- Create resilience
- Analyse monitoring results
- Engage with customer
- Development (automation or improvement)

Influence: If a player gives one or more resources for use by another player they receive one influence. The receiving player loses one influence.

If multiple players give resources to a single player, the receiving player loses only one influence in total and each of the donating players receive one influence.

If multiple players contribute to the same incident on a shared service, no influence is received.

PERFORM actions

According to your planning, perform the associated actions in whichever order you see fit. All players perform their actions at the same time.

NOTE: the result of one action cannot impact another, during the same round.

Influence: Any player that, acting alone, performs an action in support of a shared service, receives one influence.

Check resilience

Check to see if resilience has been created that would affect any of the incidents currently visible. If this is the case, remove these incidents.

Take any business value points

Successfully completing service improvement activities will generate business value.

Any completed automation cards are flipped to their "active" side and the discs are put back into your supply.

If a customer improvement request has been completed, return the discs to your supply, take the

business value shown bottom right of the card and flip the card.

If the backlog improvement issue has been completed, return the discs to your supply and take the single business value point.

If any player reaches or exceeds 20 business value points the game ends at the end of this phase. Refer to END OF GAME on page 12.

Business value is tracked along the bottom of the centre board.



Recover resources

Recover any remaining resources and place them back into the resources area.
If there are no more cards left on your operations deck, the game is over. Refer to END OF GAME on page 12.



If, at the end of a round, any service has more than 5 incidents, the game is over for all

END OF GAME

There are four ways that the game can end:
One or more players fall below -7 influence points. In this case the game is over for all. There are no winners when the one or more players are allowed to lose so much influence, sorry.

A services has 5 or more incidents. Once again, no scoring, you all lost, sorry.
One or more players have reached or exceeded 20 business value points or there are no more operations cards. Time to check the scores - see below

End of game scoring

Resilience

Each player receives an additional 1 business value point for each area of resilience created.



Influence

Adjust your business value points by one business value point for every 4 influence points. Round down for positive influence, round up for negative influence.



So, an influence of -2 would lose one business value point whereas an influence of +2 would gain none.

Your service

You gain or lose points based on your success at resolving incidents in your service:

If you have failed your support promise on any incident you lose 2 business value points.

If there is no support promise failure score the following:
0-1 incidents (no P1 incidents), 2 business value points
2-3 incidents (no P1 incidents), 1 business value point
P1 incidents, -1 business value points per incident.

If, at the end of the scoring, two or more players are tied for first place, the winner is determined in the following order:

- most resilience
- most knowledge/know errors
- most influence
- Lowest remaining incidents value; P1=3, P2=2, P1=1

If it is still a tie, there is no outright winner. Truly you equally contributed to the success of your services and, by extension, the company. Well done!



Solo play

If you have scored 10 or more points you can be quite satisfied with your performance. More than 15 points is excellent.

ACTIONS

FIX INCIDENT

Occasionally things stop working as expected. It may be a complete failure or it could just be a reduction in performance below agreed thresholds. In either case, an incident will be created to identify the issue and route it to the appropriate team for resolution. Incidents can be categorized according to their urgency, such as priority 1, 2, or 3. Each priority level will have an agreed support promise, detailing the

amount of time before the incident must be fixed. If the incident is not fixed in the allotted time, the support promise has been broken.

An incident may be of a known error type, which makes it easier to fix, consuming fewer resources, as there is less investigation time spent identifying the error.

Fixing an incident

It takes two resources to fix an incident. The first resource is used to identify the issue and the second resource resolves the issue.

If the error type has been documented as a known error, the first resource is not needed.

NOTE: If the same incident is present more than once, fixing one will fix all.

DOCUMENT KNOWN ERROR - Knowledge article

Knowledge management is fundamental to ensuring that everyone has access to, and an understanding

of, information relevant to the task in hand.

Document known error

This can only be done as an extension to a "fix incident" action and by the same player. If the incident has been fixed by more than one player, either can choose to create a knowledge article documenting the error. Adding a knowledge article, documenting the incident type and resolution as a known error, will reduce the effort needed to resolve any future similar issue. It will also pave the way for a future root cause analysis to identify and remove the cause of the error, thereby creating resilience.

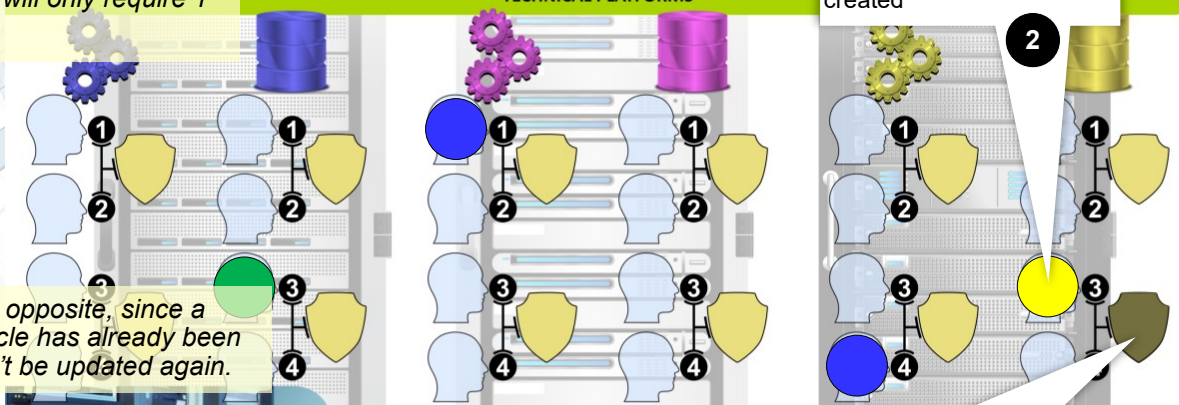
It costs one resource to add the knowledge article, placed during the ASSIGN phase.

During the PERFORM phase, the player places one of their discs onto the known error location corresponding to the error type. This can only be done if the spot is initially empty and resilience has not been created at the node.

Once an incident is fixed, the card is removed from play.

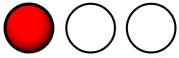
Here, the Known error has been updated by the yellow player (2) so the incident will only require 1 resource to fix.

In the example opposite, since a knowledge article has already been updated, it can't be updated again.



Resilience here would have avoided the incident shown in the example opposite

Priority 1 (P1)



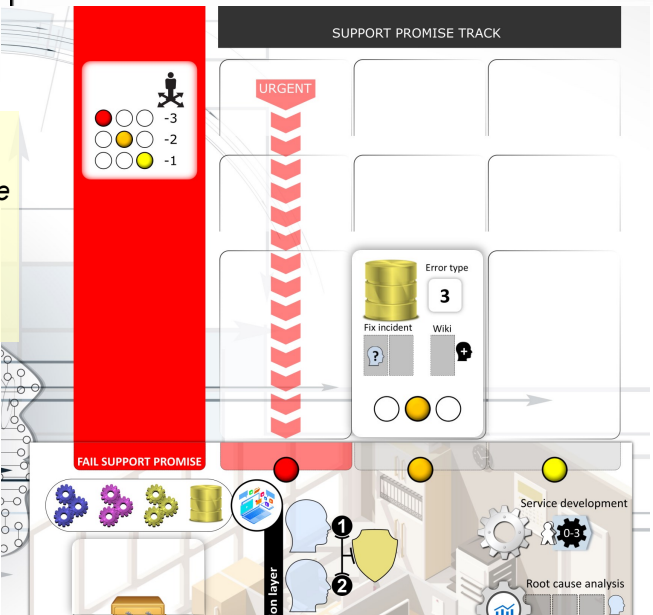
Priority 2 (P2)



Priority 3 (P3)



In this example, if there had been a disc on the resilience location associated with error type 3, of the yellow database platform (see opposite), the card would not generate an incident as the error type would have been removed.



CREATE RESILIENCE - Root cause analysis

Problem management sets out to reduce the number of incidents. One aspect of problem management is to further analyse known errors to establish the root cause, which, once addressed, creates resilience.

Helen is playing the blue colour and has the Administrative service to manage. Previously, she has fixed an incident, caused by error type 1 in the application layer, and has created a known error knowledge article (1).

She now decides to perform a root cause analysis to increase the resilience in her service.

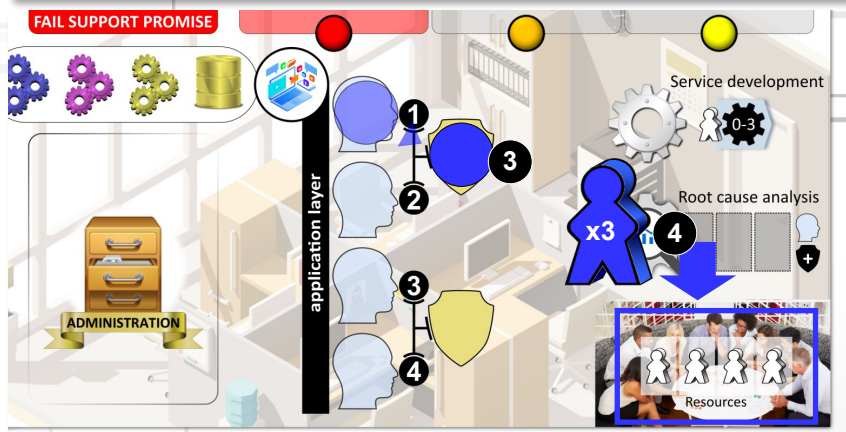
In the ASSIGN phase, she places 3 resources in the root cause analysis area to the right of her board (2).

Now it is time to PERFORM the action. Since Helen has previously documented the known error herself, she moves that disc to the right, creating resilience (3). If it had been someone else's disc, it would have been returned to its owner and Helen would place one of her own discs onto the resilience space.

If there had been a disc on the known error knowledge article for both error type 1 and error type 2, one of these would be returned to its owner, leaving both spaces empty.

Finally she recovers her resources and places them back into the resource area (4).

Once a root cause has been performed, review all incidents in play and remove any that have been affected by the newly created stability.

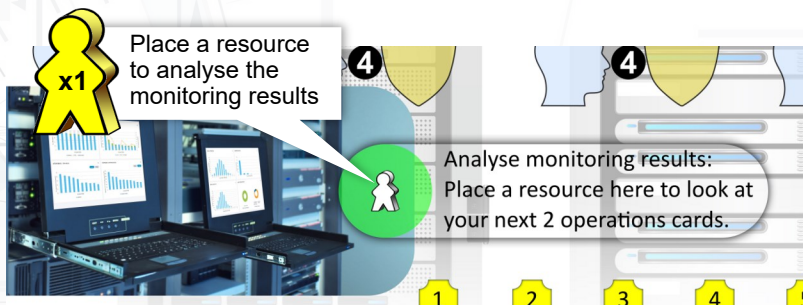


ANALYSE MONITORING RESULTS - Predict future events



A wealth of information can be gathered from the monitoring results if you know how to analyse them.

Jean really wants to know what events are on their way. During the ASSIGN phase, she places a resource on the monitoring section below the technical platforms. This allows her to look at the next two event cards. She then returns them to her operations deck, without changing their order.



ENGAGE WITH THE CUSTOMER - Customer improvement requests



The continual improvement register represents a backlog of identified service improvement. These may have come from the DevOps team or been received from the customer/user. Since the customer really understands the business impact of the requested improvements, any requests

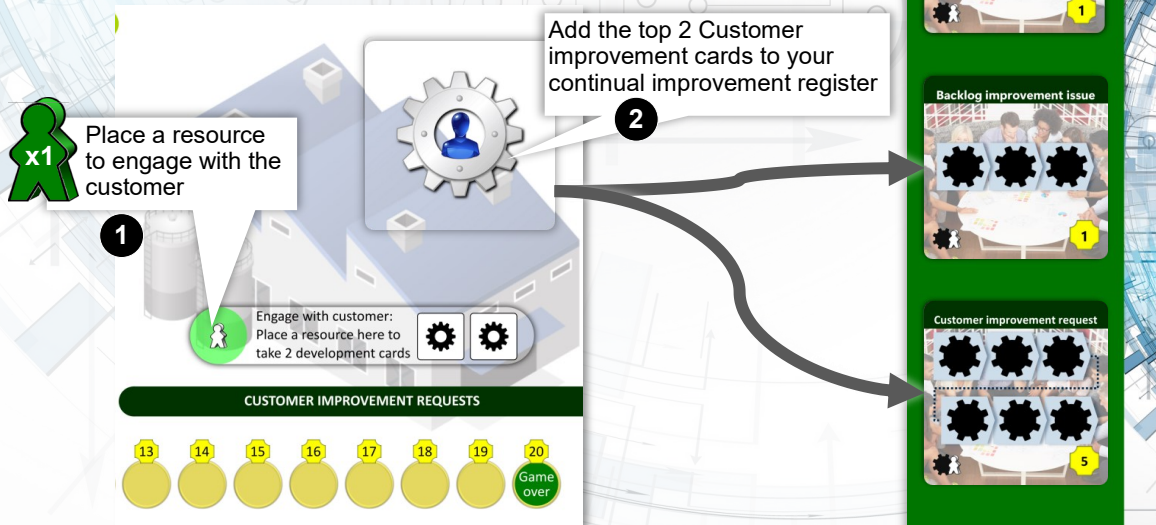
received from them will generate more benefit to the business than those identified independently by the DevOps team. However, time must be spent engaging with the customer, to learn what improvements are considered important.

Engaging with the customer

Phil decides that it is time to find out what the customer wants improving in his manufacturing service. In the ASSIGN phase, Phil places a resource in the customer improvement requests area on the central board.

In the PERFORM phase, he takes the top two cards from the customer improvement deck and adds them to his continual improvement register area.

A card can be placed on an empty location, on a previously flipped improvement, or replace any existing card. If an existing card is replaced, any discs on that card are returned to the supply, and the replaced card is shuffled back into the customer improvement deck.



DEVELOPMENT



During the **ASSIGN** phase, up to three resources can be placed on available development steps, for request automation, continual improvement, or both.

During the **PERFORM** phase, these resources are returned to the resource area and replaced with discs

to show that the step has been completed. Any completed cards continual improvement cards are scored.

Where the development tracks are completed, the discs are returned to the supply and the card flipped, except for the backlog improvement issue.

Developing request handling automation

Phil decides to develop his request automaton and chooses to use 2 resources for development. Phil had previously developed 2 steps of "Automation of request #1" (1), leaving just 1 step needed to complete the automation. He places one resource on the third development step (2), and places the other on the first development step of "Automation of request #2" (3).

Perform development work

In the **PERFORM** phase, the resources are returned to the resource area and replaced by discs (4).

Since the automation of request #1 is now completed, the discs are removed and returned to his supply, and the card is flipped to show that the automation is active (5).

In future, if he gets a request #1 on the service desk roundel there will be no need to assign resources.



Developing continual improvement

Susan decides to assign the maximum 3 resources to continual improvement.

She has previously developed one step in the standard backlog improvement issue (1) but has since added two customer improvement requests to her continual improvement register.

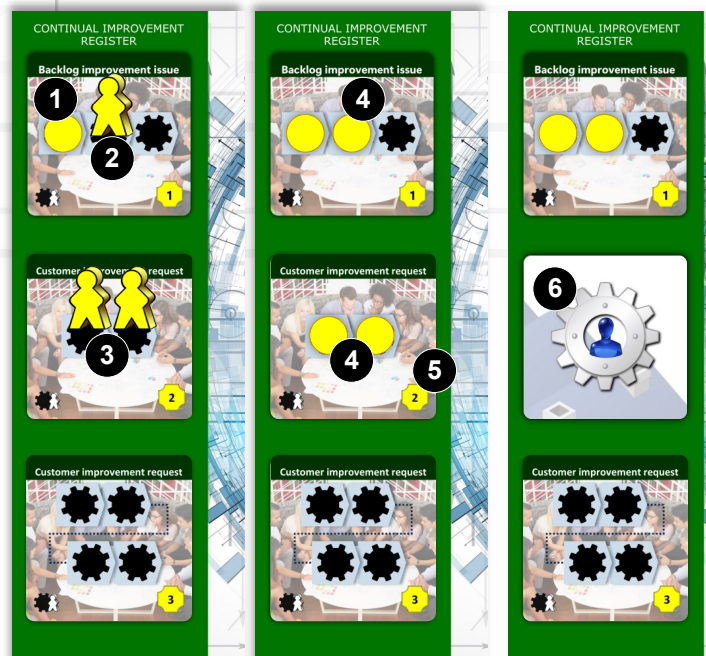
She has several options as to how to place the resources, and decides to place one resource on the second development step of the backlog improvement issue (2). She places the other two onto the first two development steps of the upper customer improvement request (3).

Perform development work

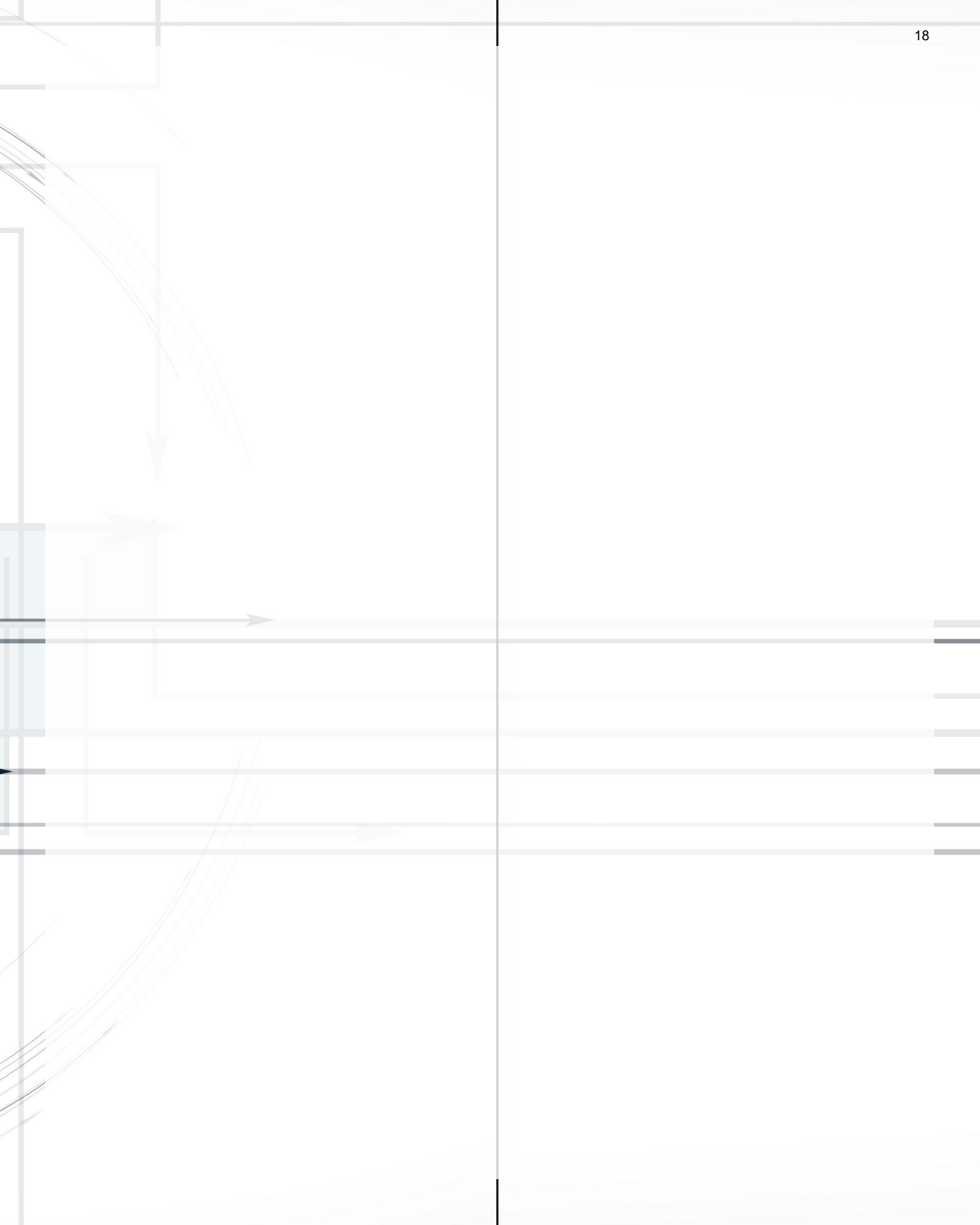
In the **PERFORM** phase, the resources are returned to the resource area, and replaced by discs (4).

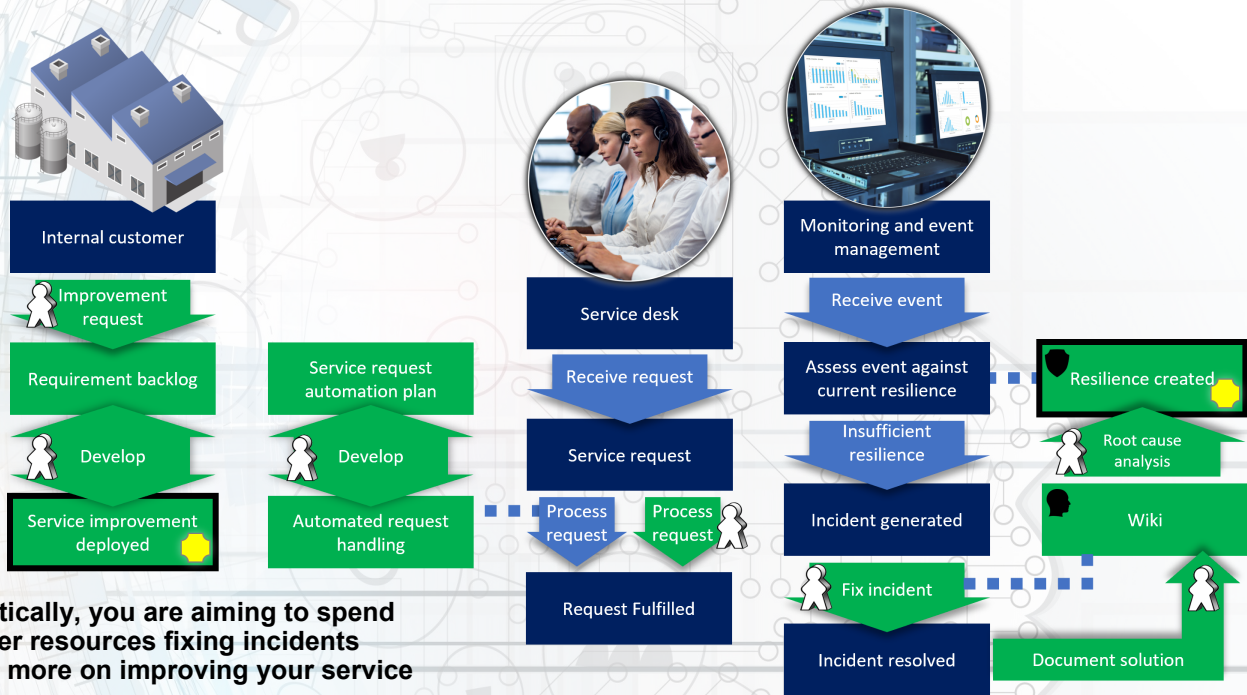
Take business value points

The upper customer improvement request has now been completed, with both development steps containing a disc. She earns 2 business value points for this development, as indicated bottom right of the card (5), which she takes by advancing her disc on the centre board's business value track. Susan returns the discs to her supply and flips the card (6).









Tactically, you are aiming to spend fewer resources fixing incidents and more on improving your service

If anyone goes below an influence of -7 the game is over for all
If, at then end of a round, any service has more than 5 incidents, the game is over for all

End-of-game scoring

Resilience

Each player receives an additional 1 business value point for each area of resilience created.



Your service

You gain or lose points based on your success at resolving incidents in your service:

If you have failed your support promise on any incident you lose 2 business value points.

Influence

Adjust your business value points by one business value point for every 4 influence points. Round down for positive influence, round up for negative influence.



If there is no support promise failure score the following:
 0-1 incidents (no P1 incidents), 2 business value points
 2-3 incidents (no P1 incidents), 1 business value point
 P1 incidents, -1 business value points per incident.

So, an influence of -2 would lose one business value point whereas an influence of +2 would gain none.

Resilience is the tie breaker, followed by Knowledge, Influence and Remaining incidents value.