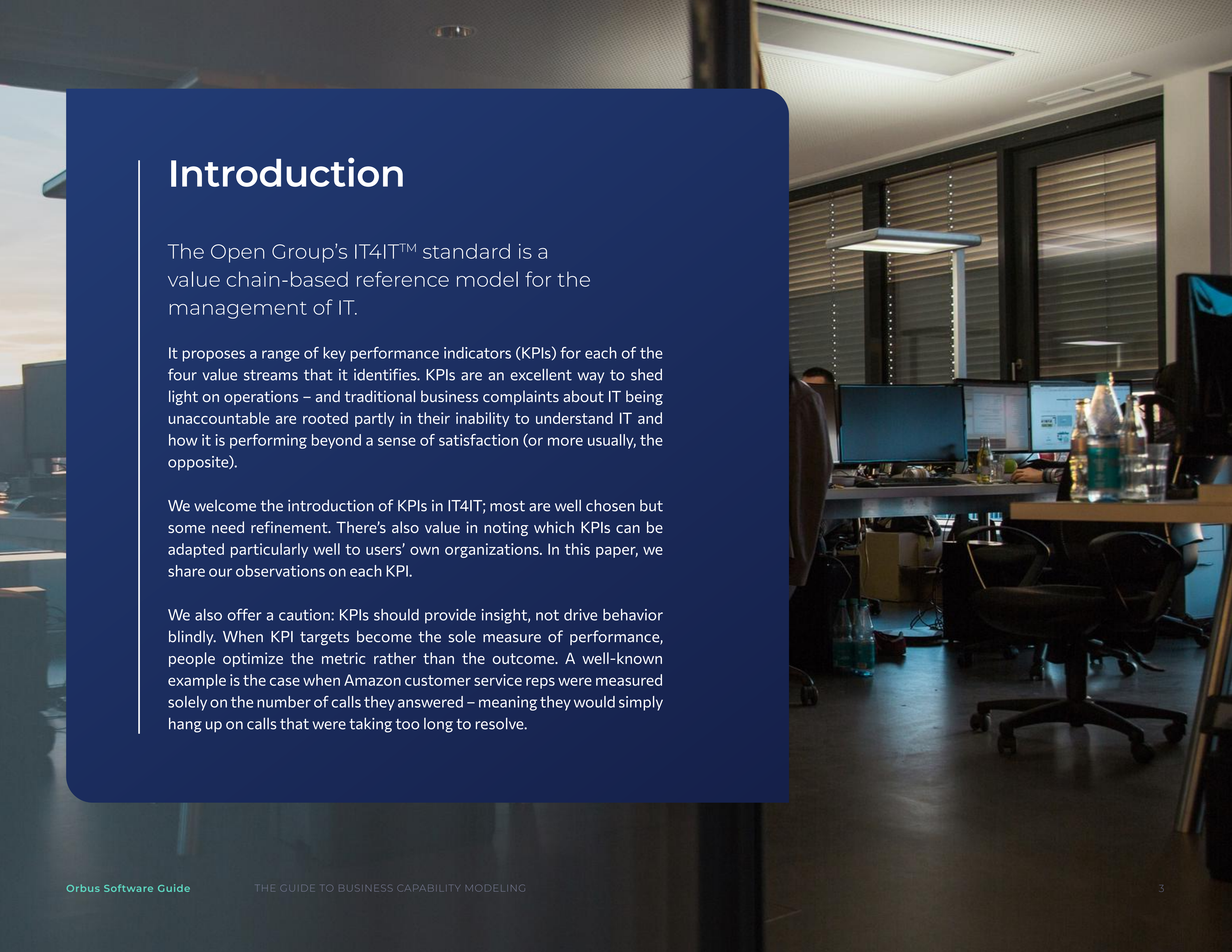
OrbusSoftware

An analysis of **the IT4IT KPIs**

Contents



Introduction

The Open Group's IT4IT^{TM} standard is a value chain-based reference model for the management of IT.

It proposes a range of key performance indicators (KPIs) for each of the four value streams that it identifies. KPIs are an excellent way to shed light on operations – and traditional business complaints about IT being unaccountable are rooted partly in their inability to understand IT and how it is performing beyond a sense of satisfaction (or more usually, the opposite).

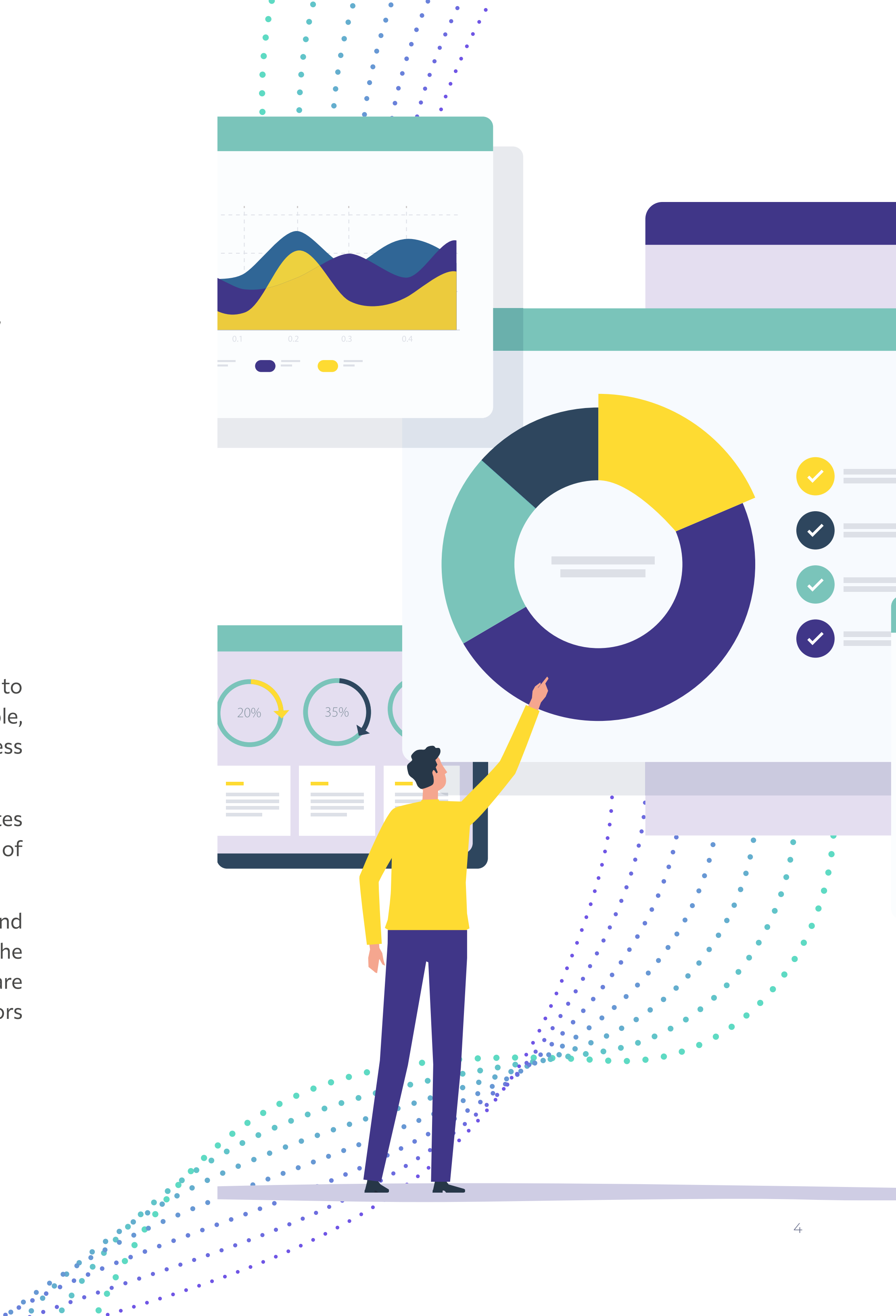
We welcome the introduction of KPIs in IT4IT; most are well chosen but some need refinement. There's also value in noting which KPIs can be adapted particularly well to users' own organizations. In this paper, we share our observations on each KPI.

We also offer a caution: KPIs should provide insight, not drive behavior blindly. When KPI targets become the sole measure of performance, people optimize the metric rather than the outcome. A well-known example is the case when Amazon customer service reps were measured solely on the number of calls they answered – meaning they would simply hang up on calls that were taking too long to resolve.

Choosing Key Performance Indicators

When we're defining key performance indicators, we need to start by defining what performance actually is. A standard method is to define Critical Success Factors (CSFs) – the things that the organization needs to be doing. One good reference is the paper "A Primer on Critical Success Factors" by Christine Bullen and John Rockart. They identify three aspects of critical success factors.

- **Internal vs External:** whether the critical success factor relates to an area wholly under the organization's control or not. For example, interaction with customers would be an external critical success factor.
- **Monitoring vs Building:** whether the critical success factor relates to improving existing operations versus implementing some form of change.
- **Source:** where the critical success factor comes from. Bullen and Rockart identify five sources – the industry itself, the strategy of the organization, environmental factors, critical success factors that are derived from a specific role, and short-term critical success factors that exist temporarily to respond to an unusual event.



The next step in identifying key success factors is defining how these CSFs can be measured. Now for some CSFs it may seem impossible to measure them at first glance. How can we measure something like ‘improved agility’? In any case where this becomes a problem, the important question to pose is “Why do we even care?” In other words, why do you want increased agility? What is it that you hope will change as a result of this? The desired results provide the proxies that you can use to measure the underlying (but not directly observable) quality.

Some sources suggest to apply the SMART approach to defining KPIs.

- **Specific** – target a specific area for improvement.
- **Measurable/Ordinal** – quantify or at least suggest an indicator of progress. The opposite is a qualitative KPI, that merely requires that something exist... the issue being that such a KPI cannot show any change.
- **Assignable** – specify who will do it.
- **Realistic** – state what results can realistically be achieved, given available resources.
- **Time-related** – specify when the result(s) can be achieved.

As with the use of KPIs themselves, the one caveat here has to be that these five aspects of a KPI should guide the definition of a KPI – they should never be applied if they harm the intent of the KPI, which is to gain insight into how a particular success factor is changing. Also, the cost of collecting the information for a KPI should be small – otherwise the cost incurred by the measurement outweighs the benefit of having it.

With these observations made, we’re ready to look at the KPIs that IT4IT offers.

Strategy to Portfolio (S2P) Value Stream

KPI	Analysis
Business and IT Alignment	
Ratio of new versus maintenance services.	Our first KPI in the set presents some issues. First, the definition of what a new and what a maintenance service is, is unclear – at what point does a new service become a maintenance service? Immediately on implementation? After three months?
Accurate Visibility into Overall Demands from Business	
Demand requests, types, and delivery per service % of overall IT budget that can be traced to formalized demand requests.	An excellent KPI - it shows an unambiguous quantitative measure.
Structured and rationalized Demand Management with ongoing efforts to minimize the number of demand queues that staff must respond to.	The second sentence listed for this CSF is really a separate KPI. This KPI is a littleambiguous but it's purely qualitative. However, the definition does suggest an improvement in the description – monitoring the number of demand queues would provide a quantitative KPI.
Service Portfolio Rationalization	
A formal Service Portfolio functional component process exists under the ownership of the Service Portfolio Management process owner.	This is another qualitative KPI in that it's a checkbox – either it happens, or it does not. Although two operations are described, it's still just one KPI as producing the accounting records without using them is pointless
Taxonomies for understanding functional and technical redundancy and business value of the IT service are implemented.	
Processes for consistently evaluating and tagging portfolio entries are implemented.	

KPI	Analysis
Service portfolio is subject to ongoing rationalization using the taxonomies, implemented as continuous improvement. Service and IT Portfolio Management are themselves rationalized with clear scopingand relationship established.	
Service Portfolio Financial Analysis	
Accounting records are produced on a regular basis to show the ongoing "investment & spend" in each service/application. These are comparedwith business outcomes and financial objectives that have been achieved.	This is another qualitative KPI in that it's a checkbox – either it happens, or it does not. Although two operations are described, it's still just one KPI as producing the accounting records without using them is pointless.
Service Portfolio Reporting and Analysis	
A service portfolio exists and is used as the basis for deciding which services to offer.	This presents another qualitative KPI that, alas deals with existence of process, making it hard to provide a quantitative KPI.
Service Investment Tracking	
The investment in each service is quantified in the service portfolio.	This is a qualitative KPI, but in truth it deserves to be a quantitative KPI – the percentage of services for which investment is quantified. The reason we make this distinction is that some services are probably easier to quantify than others – so a single 'on/off' measure makes ittempting to accept wild estimates.
Investment in each service is reported, starting with the initial investment, and followed by monthly, quarterly, or annual reporting of the ongoing budget spend (total cost of ownership).	A qualitative KPI, although it could be made into a quantitative KPI after the first period. The actual spend could be tracked against planned spend, or (more usefully) historical spend into order to give insight into either seasonal cost of service provision or yearly trends in cost bases.

KPI	Analysis
Improve Customer Satisfaction	
Satisfied customers per service/application.	The description of this KPI seems to imply that there is a customer satisfaction survey that is sent out for each service... which offers up two considerations.
Stewardship of IT Investment	
Software license percentage in use.	At first glance, a nice quantitative and unambiguous KPI. However, it requires two areas of implementation. First, accurate tracking of licenses needs to be in place. Second, it does ignore enterprise licenses. A possible KPI for these would be to track users per enterprise license.
Planned versus actual service costs.	Here the KPI is unambiguous and quantitative. However, it does require an estimation of service costs. So until data exists actual implemented service costs, tracking this KPI should be hedged with high leeway on disparity of estimation.
Average cost of IT delivery (per service/ application) per customer.	Again, a quantitative KPI that faces issues with accurate information collection. There will need to be a means to match not only services to customers, but the cost per service per customer. Careful consideration of how this could be achieved is necessary before implementing this KPI.
Enterprise Security Alignment	
Frequency of security assessments against latest standards and guidelines.	This is a nice unambiguous KPI that also meets an important test – low cost of collection.
Noted deficiencies against security standards and policies.	Again, a good quantitative KPI; however it might be interesting to break it out by minor and deficiencies similar to common practice in audits.

Requirement to Deploy (R2D) Value Stream

KPI	Analysis
Improve Quality	
Number of escaped defects	As with the previous value stream, the first KPI seems good in intent but is harmed by ambiguity. From the context we would assume that 'escaped defects' means 'defects present in the first release to customers'. Escaped here would mean 'that didn't get released – it escaped'. This KPI would normally be broken out by service and but there may also be value in breaking it out by service grouping.
% of actual versus planned executed tests	Tracking actual versus planned tests can be useful, but the one worry with this measure is that it probably should be seen as a success measure. If a test shows that an entire set of subsequent tests cannot be executed, there should be no penalty for simply cancelling them.
% of critical defects found early in unit testing versus UAT	Unfortunately, this KPI has an Achilles heel – what is 'early'? From the readingand the context, we suspect that the intent is simply to compare defects in unit testing versus UAT. It does, however, ignore other stages of testing such as integration testing. Given that bug tracking software is so widespread, % of defects found at each stage of testing would be a useful measure.
Improve Project and Feature Execution	
% of projects (project tasks, stories, other demand requests) on time	A fairly standard, even traditional projectmeasure.
% of healthy projects (projects without unresolved urgent issues)	As with several other KPIs that IT4IT proposes, this is a reasonable KPI assuming that definitions are in place. Specifically, definitions of 'unresolved' and 'urgent'.

KPI	Analysis
Deviation of planned to actual work hours	A fairly standard, even traditional project measure.
Number of identified issues	Again, a standard KPI, but it would normally be broken out by some measure of importance.
Number of opened risks	Again, a standard KPI, but it would normally be broken out by some measure of importance and potential impact.
Amount of backlog/work-in-process	A fairly standard, even traditional project measure.
Arrival and departure rate for work	This KPI suffers from a slightly unclear definition in that ‘work’ is not defined. It could be taken as the number of requests, but perhaps a better measure would be the number of function points aggregated over user stories for each request.
Improve Stewardship of IT Investment	
% of actual versus planned project cost	A good, unambiguous project measure that faces one issue – the fear of being wrong. We’ve stated earlier that KPIs can only be one way of gaining insight and this is particularly true of this KPI. Until enough historical data is in place, inaccurate estimates should not be seen as a failure.
% of change in project cost	A fairly standard KPI – this one is probably best used to look for trends between and amongst projects.
% of budget at risk	The one issue with this KPI is defining the percentage of budget at risk. We could define it as a sum of the weighted risk amounts for the overall IT risk register; however, the question with this KPI is ‘what would this information be used for’?

KPI	Analysis
Increase Automation Adoption	
% of automated tests	A pretty good KPI in that it is unambiguous; however, it runs the risk of been seen as a target; as stated previously, setting targets for KPIs needs to be taken with caution. For example, setting a target for % of automated tests disincentives manual testing. A complementary measure would be number of automated tests. Both KPIs would normally be broken out by project.
Achieve Development Process Excellence	
% of requirements tested, authorized, completed	A reasonably standard requirements KPI with no issues.
% of requirements traced to tests	A reasonably standard requirements KPI with no issues.
% of reviewed requirements	A reasonably standard requirements KPI with no issues.
% of successful builds	A reasonably standard requirements KPI with no issues.
% of changes resulting in Incidents	Here there is one problem – knowing for sure that a specific change resulted in an incident; correlation does not always mean causation. Of course, this matching could be done manually. Hence, this is a KPI that needs to be taken in the context stated above.
Ratio of detected to closed defects at release	A reasonably standard development KPI with no issues.
Improve Early Life Success of Releases	
% of Incidents during warranty period	A reasonably standard release management KPI with no issues.
% of successful/unsuccessful deployments for the project	Here there is one issue – the definition of 'unsuccessful'.
% of emergency changes	A reasonably standard release management KPI with no issues.
Pass rates on UAT/validated requirements	A reasonably standard release management KPI with no issues.

KPI	Analysis
Operations and Development Collaboration	
Trend on early life support/UAT success metrics	Unfortunately, this is not really a usable KPI in that it isn't really defined.
% rework	While this KPI seems simple at first glance, questions arise on how to calculate it. The most sensible approach would seem to be the amount of time spent on rework.
Improve Financial Visibility	
Planned cost versus actual cost	As with the other 'planned versus actual' metrics, the caveat we would give to someone implementing this KPI is that you can initially expect some wild variance in estimation – so setting targets for this KPIin the initial stages is inadvisable.
Maintain a Linkage between Business Services and IT Initiatives	
Aggregate (roll up) service development costs by business service	A good KPI with no issues.
High Quality Service Design Specifications at the Outset	
% reduction in the rework required for new or changed service solutions in subsequent lifecycle stages	A reasonable KPI.
Integration Test Success	
Trend on the number of installation errors in all the packages in the integration environment	The one issue that we have with this KPI is that it is a KPI on a KPI – monitoring trends should be implicit in the management of KPIs in general.
Number of applications or services that require exceptions outside of the existing infrastructure portfolio	A reasonable KPI.

KPI	Analysis
Design-Review to Ensure Application Design Complies with all Policies, including Security	
Number of application designs that pass a security policy review	Another nice clear quantitative KPI.
Early Testing of Applications for Security Vulnerabilities	
% of severity 1 security defects fixed before application is released	An excellent KPI. We've not seen any organizations that do not classify defects, and the release of an application is an unambiguous event.

Request to Fulfill (R2F) Value Stream

As we move along the value chain into the Request to Fulfill stream, the KPIs tend to become more obvious.

KPI	Analysis
Ability to Meet Customer Expectations	
New or modified Subscriptions per time period	A good quantitative KPI that in keeping with the mindset of IT4IT does presuppose to a SaaS-based approach.
% and number of Subscription requests complying or breaching SLA or OLA agreements	A good KPI with no issues.
Number of Subscription requests accepted and rejected by the requestor for the first time right delivery/fulfillment	A good KPI with no issues.
Variation in the average time to fulfill Subscription requests for the predictability of delivery	A good KPI with no issues.
Number of Incidents related to request fulfillment	A good KPI with no issues.
Arrival and departure rate of service requests	A good KPI with no issues.
Reduce Costs	
Costs (burned resources) per service and per fulfillment step	This KPI does require a very tight understanding, not just of the cost for each service but also for each step in provision of the service. So it seems very much as an end-goal KPI rather than one that could be implemented immediately.
Breakdown of self-source fulfillments versus one-off fulfillments	Here, the only issue is precisely defining what a one-off fulfillment is.

KPI	Analysis
% and number of fulfillments requiring human intervention to be completed	A good pair of KPIs with no issues.
% and number of fulfillments requiring human intervention to be completed	A good pair of KPIs with no issues.
Number of service request queues being managed	A good KPI with no issues.

External Service Provider Compliance

KPI	Analysis
Number of purchase orders per time period	A good KPI with no issues.
% and number of orders delivered and accepted complying with underpinning contract agreements	A good KPI, but it and the following KPI effectively form a spanning set of all delivered orders – meaning that tracking both independently is superfluous.
% and number of delivered orders breaching underpinning contract agreements	As described above.
Number of Incidents related to the purchase order fulfillment	A good KPI with no issues.
Number of purchase orders unfulfilled at the end of a given period	A good KPI with no issues.
Number of orders delivered and accepted by the requestor per time period	A good KPI with no issues.
Number of purchase orders rejected via no delivery or cancelled purchase orders	A good KPI with no issues.

KPI	Analysis
Increase Speed/Agility/Flexibility (Operational Performance)	
Completed service requests	A good KPI with no issues.
Service request work-in-progress	A good KPI with no issues.
Number of interactions with consumers per service during delivery	This KPI seems reasonable, but it will require a very disciplined approach that requires all interactions to be logged in order to provide accurate statistics.
% of work-in-progress within SLA	A good KPI with no issues.
% of completed work within SLA	A good KPI with no issues.

Detect to Correct (D2C) Value Stream

A running theme through the Detect to Correct value stream KPIs is that they are defined as sets of goals. However, the KPIs implied by the goals are usually very clear and of high quality, as they are as they are 1) Quantitative 2) Unambiguous 3) Easily Measured

KPI	Analysis
Ability to Meet Customer Expectations	
Events: • Increase in breadth and depth of monitoring endpoints • Reduction of escalated events (viafiltering/correlation/ automated resolution) • Reduction of false positives • Reduction of the number of security events that cause business disruption.	The second, third, and fourth of these KPIs are OK, but in keeping with a running theme elsewhere the first and third KPIs depend on accurate definitions. Specifically, the ‘breadth and depth’ of monitoring endpoints need definition.
Incidents: • Incident reduction • Reduction of escalated Incidents • Reduction of false positives • Reduction in the total number of security-related Incidents	A good family of KPIs with no issues.
Problems: • Increase Problems identified • Increase Problems eradicated.	A good pair of KPIs with no issues.

KPI	Analysis
Changes: • Reduction of change-related outages • Reduction of emergency changes • Reduction of unplanned changes • Reduction of security vulnerabilities introduced during Change Management	A good family of KPIs with no issues.
Knowledge: • Increase Known Error availability (enrich Known Error database) • Increased usage	A good pair of KPIs with no issues.
Improve Customer Satisfaction	
OLA/SLA: • Reduction of failed agreements	For once, we have a D2C KPI that we're not comfortable with; in that the term 'failed agreements' is unclear. It would seem that this refers to the number of events that break the terms of an SLA/OLA, but
Availability of critical business systems: • Increase uptime • Decrease MTTR • Increase MTBF	Three good KPIs that do form a logical family.
Performance (user experience) of critical business systems: • Decrease user complaints	A good KPI with no issues.
Incidents: • Increased rate of first call resolution	A good KPI with no issues.

KPI	Analysis
Self-service: • Increased success rate for user self-fix	A good KPI, but thought will need to be applied to how this could be measured. Number of visits to self-service resources such as a knowledge base gives some indication, but it does not unambiguously show whether the issue was actually resolved.
Improve Staff Effectiveness	
Events: • Increase automatically remediated • Increase the percentage of Events correlated to a business service	A good KPI with no issues.
Incidents: • Reduction of re-opened Incidents • Increase percentage of first call resolution • Reduction in average time to close an • Increase automatically remediated • Reduce average handling time • Reduce rejected Incidents	A good family of KPIs with no issues.
Changes: • Increase automatically remediated changes	A good KPI with no issues.
Improve Staff Effectiveness	
Cost: • Increase percentage of time invested on business-critical services	A good, quantitative, unambiguous KPI with one caveat - it does require a clear definition of which business services arecritical however.
SLA/SLO: • Increase percentage of business-critical services with defined Service Level targets	Again, a good, quantitative, unambiguous KPI with the same one caveat - a clear definition of which business services are critical is needed.

KPI	Analysis
Services: • Increase number of business services defined • Decrease percentage of business-critical services • Decrease number of CIs that are not linked to a business service, • Increase "quality of service" monitoring for internal and external business services.	A good family of KPIs with no issues.
Security: • Number of security-related outages to business-critical systems, • Number of security Incidents causing financial loss, business disruption, or public embarrassment • Number of security Incidents resolved without business impact.	A good KPI with no issues.

Turn IT performance metrics into strategic business value

The IT4IT KPIs outlined in this guide provide the foundation for measurable IT performance, but their true power lies in connecting these metrics to business outcomes. OrbusInfinity helps organizations optimize technology investment, accelerate innovation, and strengthen resilience by providing end-to-end visibility across your IT landscape.

Discover how leading enterprises use OrbusInfinity to:

- Reduce IT costs by identifying redundant systems and optimizing spend
- Accelerate time to market with clear visibility into project impacts and dependencies
- Strengthen compliance and resilience through proactive risk identification
- Demonstrate ROI by tying IT investments directly to business value

Ready to transform your IT metrics into strategic advantage?

BOOK A DEMO

OrbusSoftware

Orbus Software UK
LONDON

Orbus Software USA
NEW YORK

Orbus Software ANZ
SYDNEY

Orbus Software EU
KATOWICE

Orbus Software MEA
DUBAI

Orbus Software ASIA
SINGAPORE

About Orbus Software

Orbus Software is a leading global provider of enterprise transformation solutions. We aim to empower customers with a strategic decision-making platform to successfully manage complex change. Our OrbusInfinity platform enables leaders to deliver business objectives, innovate faster, and ensure enterprise resiliency, while supporting them to make more informed, responsible, and sustainable business decisions.

orbussoftware.com