

WHITEPAPERS

Controlling changes to your key reports

The Oracle reports your controls and auditors rely on (IPEs), what sits behind them in BI Publisher and Oracle Analytics, and how to show when each one changed and that every change was requested and approved.

IN BRIEF

- Many financial controls rest on reports from Oracle Fusion: the key reports, or IPEs (information provided by the entity), that management and auditors rely on.
- A key report is more than the report people open. Its numbers come from objects behind it: a data model for a BI Publisher report; pages, analyses and prompts for an analytics dashboard.
- A change to any of those objects can change the numbers, while the report itself shows no change at all.
- Keeping track of those links by hand is impractical, and they change over time.
- Sentinel links each key report to everything behind it automatically, follows changes to every part, and matches each change to its change request.
- The result is a full change history for each report, and an evidence pack for any period.

Why key reports matter

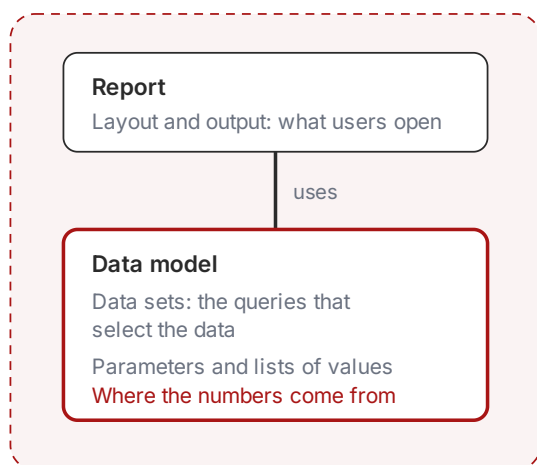
Behind many financial controls is a report. A reviewer signs off a reconciliation against a ledger report. A manager approves journals listed by an exception report. An auditor selects a sample from a population that a system report produced. In audit terms these are IPEs, information provided by the entity, and auditors need to be satisfied that they are complete and accurate before relying on them.

Once a report has been tested and accepted, the question for every later period is simple: has it changed since? If it has, was the change requested, approved and made by the right person? Seeing when each key report changed, and linking every change to its change request, is strong evidence that the report can still be relied on.

A report is more than the report

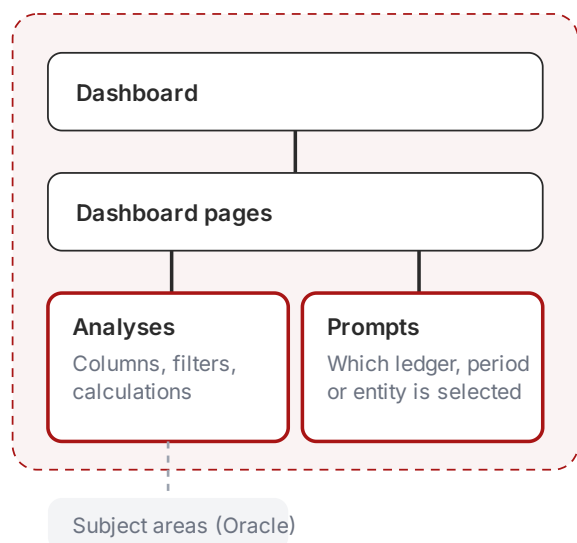
The report a user opens is only the visible part. In Oracle Fusion, the numbers it shows depend on other objects in the reporting catalogue, and those objects differ between Oracle's two reporting tools.

BI Publisher



Dashed outline: linked and tracked by Sentinel

Oracle Analytics (OTBI)



The visible report depends on other catalogue objects, and the two reporting tools are built differently.

BI Publisher reports

A BI Publisher report is a layout. The data it shows comes from a **data model**, a separate object in the catalogue that holds the data sets (typically the queries that select the data), the parameters and the lists of values. The data model is where the numbers come from.

That separation has two consequences for control. A change to the data model, such as an edited query, changes the report's output while the report itself is untouched. And because one data model can serve several reports, a single change can alter more than one key report at once.

Analytics dashboards and analyses

An Oracle Analytics (OTBI) dashboard is built from **pages**, and each page brings together **analyses** and **prompts**. An analysis defines the columns, filters and calculations it reads from Oracle's subject areas. The subject areas come from Oracle's own semantic model, which Fusion customers do not edit; the logic a customer controls is in the analyses and the prompts. A prompt decides what is selected: which ledger, which period, which business unit.

Here the logic is spread across several objects, and analyses and prompts can be shared between pages and dashboards. A changed filter in one analysis, or a new default in a prompt, changes what the dashboard shows without any change to the dashboard itself, and possibly what other dashboards show too. Where a control relies on a single analysis on its own, that analysis is the key report; where it relies on the whole dashboard, the dashboard is.

In both tools, the change that matters usually sits one or two levels below the object people call "the report".

Why this is hard to do by hand

To control a key report by hand, someone has to know every object behind it, keep that list up to date as reports are redesigned, and watch Oracle's audit records for changes to each one. With dozens of key reports, shared data models and dashboards that evolve, the list is out of date almost as soon as it

is written. In practice, control ends up resting on the report itself, which is exactly the object least likely to show the change.

How Sentinel links it all for you

With Sentinel, you choose the key report from your reporting catalogue: a BI Publisher report, an analytics dashboard, or a single analysis. You do not enter any technical paths. Sentinel reads the catalogue through Oracle's own catalogue services and works out the structure itself:

- for a **BI Publisher report**, the data model it uses;
- for a **dashboard**, its pages, and the analyses and prompts on each page.

Shared objects are recognised as shared. A data model used by three key reports is one object, and a change to it counts as a change to all three; the same goes for an analysis or a prompt used on several dashboards.

Links are made on exact catalogue paths. Sentinel never guesses from similar names, so a link it shows is a link that exists. Other items stored alongside a report, in the same catalogue folder, are listed separately as supporting elements, so you can see them without them being counted as part of the controlled structure.

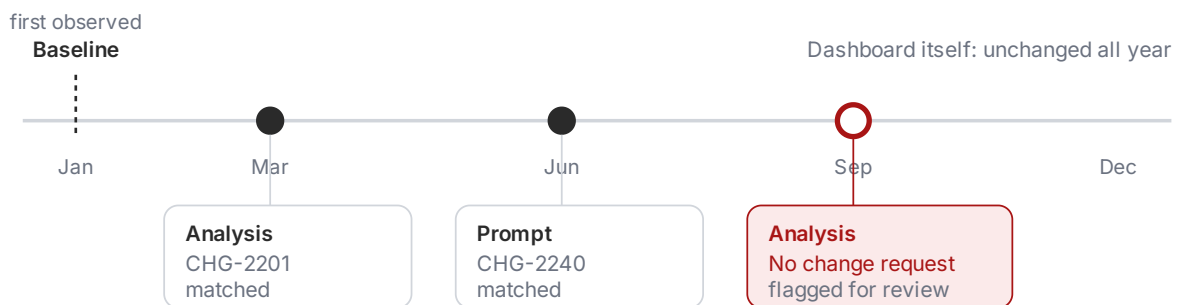
The structure is also kept over time. The structure Sentinel first observes is the report's baseline. If the report later starts using a different data model, or a page gains a new analysis, the new link applies from the moment Sentinel observed it, and is never backdated. The history therefore shows which objects made up the report at each point in the period, not just today.

Following every change

With the structure in place, Sentinel follows Oracle's own audit records for every object in it: the report, its data model, its pages, its analyses and its prompts. It detects any change to an object's definition, including the queries in a data model, and shows when it changed, who changed it and which key reports are affected. Changes are grouped into change episodes, one unit of

work by one person, and each episode is matched to the change request that explains it, in the same way as for every other change in Oracle Fusion. Every change Oracle records should link to a change request; the exceptions stand out.

IPE-07 · Journal exceptions dashboard



The dashboard never changed, but an analysis and a prompt on it did. Each change is dated and matched, or flagged.

The matching works in both directions. Sentinel also lists change requests that relate to a key report but have no matching change in Oracle, so they can be investigated too.

A full change history, ready for audit

For any period, Sentinel builds the evidence for each key report:

- a one-line result, such as “3 changes identified, 2 matched to change requests”, or a clear statement that no changes were found in the retained evidence;
- the controlled structure: each component with its catalogue path, its change history and when Sentinel first observed it, including the versions that applied during the period;
- every change episode, with its time, the person, the components changed, the change request and the review decision;
- change requests for the report with no matching change in Oracle;
- any limitation in the evidence, stated plainly rather than hidden.

The evidence pack comes as a PDF for reading and an Excel workbook for testing, with one section per report under your own control reference. Each run is kept, so earlier evidence remains available alongside later runs.

Good practice

A few habits make key report evidence stronger:

Practice	Why it matters
Give each key report your own control reference , such as its IPE number.	It names the report in the evidence pack, so auditors can tie it straight to your control documentation.
Define key reports in the production environment.	The evidence that matters is about the reports people actually rely on.
Keep key reports and their components in shared catalogue folders, never in personal folders.	A key report must be under change control, and personal folders are not. Sentinel monitors the shared folders configured for it.
Name the report, data model or analysis in the change request.	Object names in the change notes are one of the strongest signals for matching a change to its request.
Review the structure when a report is redesigned.	Sentinel picks up the new structure itself, but a redesign is a good moment to confirm the report is still the right one to control.

One point remains with you: Sentinel shows when each report changed and whether a change request backs each change. Making sure each change was properly tested before it was deployed remains your responsibility.

What this gives you

- **The whole report under control**, not just its name: data models, pages, analyses and prompts included, without maintaining the list yourself.
- **The difference between the tools handled for you**, whether a key report is a BI Publisher report, a dashboard or a single analysis.
- **A dated history of every change**, with the structure that applied at each point.
- **Every change linked to its change request**, or flagged for a recorded decision.
- **An evidence pack for any period**, in PDF and Excel, under your own control references.

This paper describes general practice and our own experience. It is not legal or audit advice; your auditors will confirm what they expect for your organisation. See also [Proving change control in Oracle Fusion](#), which covers change episodes and matching in more detail.