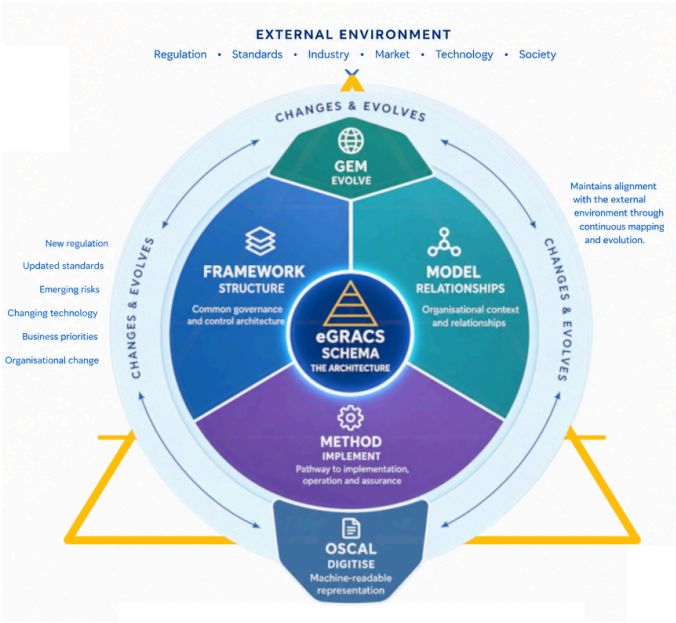
The eGRACS™ Architecture

One architecture for governance, stable at its core and adaptable at its edges.



The eGRACS architecture: one whole, six connected components.

The problem

Every new regulation tends to arrive as its own project, with its own controls, owners, evidence and audit cycle. Frameworks accumulate faster than anyone can integrate them, and the organisation ends up governed by a pile of disconnected rulebooks instead of one coherent structure. Change then becomes a rewrite.

"One architecture. Six connected components. Not another rulebook."

What it is

eGRACS gives that structure a single architecture. It does not replace the standards and regulations an organisation must satisfy; ISO 27001, SOX, GDPR and the rest keep their own authority. Instead, eGRACS provides one common structure those requirements can be related to, one way of connecting that structure to how the organisation actually works, one pathway for putting it in place, a machine-readable form, and a defined way for it to evolve.

About this guide

eGRACS — short for enterprise Governance, Risk, Audit, Compliance, Security — is a unified structural governance architecture. It is the whole: six connected components that together give an enterprise one consistent way to organise governance, risk, assurance, compliance and security, and to keep it aligned as regulations, technology and the organisation itself change.

This guide is the overview. It explains what eGRACS is, how the six components fit together, and where to go next. Each component also has its own short guide.

eGRACS™ is proprietary IP of Eljay Infotech. This summary is free to share and distribute.

How the six components fit

Component	Role	What it does	The question it answers
Schema	Orient	How the components relate to one another	*Where do I start, and how do the pieces fit?*
Framework	Structure	One common control structure: 120 controls across four tiers and three domains	*Why does every new rule feel like a new rulebook?*
Model	Connect	Relates requirements and controls to how your organisation actually runs	*How do requirements become how we actually run?*
Method	Implement	A pathway to implement, evidence and assess	*Does this mean starting over?*
OSCAL Catalog	Digitise	Governance in machine-readable form, built on NIST's OSCAL	*Why does our governance live in documents only people can read?*
GEM	Evolve	Evolves governance from feedback, lessons learned and external change	*What happens when the rules change again?*

Schema describes how the components relate. Framework, Model and Method are the three core components; the OSCAL Catalog and GEM support all of them. The OSCAL Catalog is built on NIST's OSCAL; reference to NIST's OSCAL does not imply NIST endorsement.

Ways to start

Organisations do not have to start over. Three application modes suit different starting points:

Replacement	eGRACS becomes the primary internal governance structure, for fragmented governance. This concerns internal arrangements only; external obligations still apply in full.
Overlay	A unifying layer over practices that already work.
Translation Layer	A common language across frameworks that must stay in place.

What eGRACS is not

It is not a replacement for external standards or regulations, and it is not a certification or a product. Mapping a requirement to a control shows where it relates; it does not by itself show that the requirement is met. Where a requirement is not covered, eGRACS shows the gap instead of hiding it.

Who this is for

CISOs, risk, audit and compliance leads, and enterprise architects who have been asked "what does our governance environment actually look like?" and struggled to give a straight answer.

One architecture, stable at its core and adaptable at its edges.

Where to go for more

01/06 • SCHEMA	**02/06 • FRAMEWORK**	**03/06 • MODEL**
Orient	Structure	Connect
04/06 • METHOD	**05/06 • OSCAL**	**06/06 • GEM**
Implement	Digitise	Evolve

eGRACS™ is proprietary IP of Eljay Infotech. This summary is free to share and distribute.