## Definition

A constraint places restrictions on the design or implementation choices available to the developer.

## Purpose

Constraints can be imposed by external stakeholders, by other systems that interact with the one you’re building or maintaining, or by other life cycle activities for your system, such as transition and maintenance.

Other constraints result from existing agreements, management decisions, and technical decisions (ISO/IEC/IEEE 2011).

## Sources of constraints

* Specific technologies, tools, languages, and databases that must be used or avoided.
* Restrictions because of the product’s operating environment or platform, such as the types and versions of web browsers or operating systems that will be used.
* Required development conventions or standards. (For instance, if the customer’s organization will be maintaining the software, the organization might specify design notations and coding standards that a subcontractor must follow.)
* Backward compatibility (tương thích ngược) with earlier products and potential forward compatibility, such as knowing which version of the software was used to create a specific data file.
* Limitations or compliance requirements imposed by regulations or other business rules.
* Hardware limitations such as timing requirements, memory or processor restrictions, size, weight, materials, or cost.
* Physical restrictions because of the operating environment or because of characteristics or limitations of the users.
* Existing interface conventions to be followed when enhancing an existing product.
* Interfaces to other existing systems, such as data formats and communication protocols.
* Restrictions because of the size of the display, as when running on a tablet or phone.
* Standard data interchange formats used, such as XML, or RosettaNet for e-business.

## Handling quality attributes on agile projects

* It can be difficult and expensive to retrofit (trang bị thêm) desired quality characteristics into a product late in development or after delivery.
* That’s why even (đó là lý do tại sao ngay cả) agile projects that develop requirements and deliver functionality in small increments need to specify significant quality attributes and constraints early in the project.
* This allows developers to make appropriate architectural and design decisions as a foundation for the desired quality characteristics.
* Nonfunctional requirements need to have priority alongside (dọc theo) user stories; you can’t defer their implementation until a later iteration.
* The quality requirements can span multiple stories and multiple iterations.