



CHARISMATEK Software Metrics Services – Scope Management

Scope indicates the limits or boundaries of something.

The control of scope is seen as a key factor in the control of project costs and schedule. A clear and complete understanding of a software project's scope from the project's initiation through to its completion is one key to project success.

Very often, however, when a project is long overdue or has exceeded budget, 'increase in scope' is given as the prime reason and unanticipated cause.

Given that a project will typically include a statement of scope, stating clearly what is "in scope" and "out of scope", why do so many projects run into problems of 'increase in scope'?

CHARISMATEK has investigated this issue over many projects and many years.

The most common reason - in fact, the almost universal reason - is that the items being named as the basis for scope and being tracked throughout the project have no direct correlation with effort, cost or schedule.

CHARISMATEK identifies and tracks a specific type of project deliverable at a level of granularity which has a known, well understood, relationship with effort. These are the features and functions that are to be delivered by the project, described in language easily understood by both the client commissioning the software and the software developer.

These deliverables form a negotiated and agreed statement of scope at project initiation and are monitored throughout the project. Any change can be immediately accessed for likely impact on cost and schedule.

CHARISMATEK provides a range of value adding services to software developers and purchasers focusing on effective software scope management.

See below for CHARISMATEK's service descriptions and the benefits they provide.

Scope Specification

CHARISMATEK builds a standardised model of project scope which clearly baselines the scope of the functionality and features to be delivered by the project. It is based on project artefacts available at that point in the software development life cycle.

Important features of the model are:

- ▶ The use of straight-forward business language - so that it is easily understandable by all project stakeholders;
- ▶ Cross-referencing to other project artefacts to ensure traceability and allow substantiation;
- ▶ Decomposition of the project functionality to a consistent level of granularity where the identified items have a known, well-understood relationship with effort;
- ▶ A structure which reflects any existing project specifications or requirements frameworks, making it easy to navigate;
- ▶ Ease of update to reflect any change to scope during the project to enable early identification of scope blowout;
- ▶ Its use as a key artefact for supporting other scoping activities as well as activities such as project estimation and budgeting.

Furthermore, the model of the scope will also indicate areas of uncertainty. These may or may not be able to be resolved at this point in the life cycle - but the highlighting of these areas means that, at least, you can aware of what needs further clarification.

Scope Clarification

Incomplete and conflicting requirements and design specifications will lead to significant project delays and rework.

Clients and developers make use of models of project scope to confirm and clarify their understanding of a project's deliverables.

Typically, large software development projects start with a high level expression of client needs to be satisfied by the software. These requirements may be assembled by the client prior to any developer involvement and are often quite unstructured and disorganised. There is a tendency to 'unevenness' - ranging from the very precise (prescribing the solution) to very vague (inviting a solution). The requirements document may be structured but more typically is a collation of outcomes from workshops, focus groups and the like, with contributions from multiple stakeholders.

Nevertheless, these requirements are passed to suppliers for quotation.

CHARISMATEK can collate these requirements in a model of project scope, identifying all the functions and features to be delivered by a software project. The use of a standardised level of granularity in conjunction with straight-forward business language to name individual functions ensures clients and developers have a common, shared understanding of what is to be delivered.

Conflicting or incomplete or unclear requirements are highlighted. Any areas where there are differences of opinion are quickly found and the differences are resolved. Traceability to project requirements and design specifications ensures that those specifications are also easily updated.

Good communication and a common understanding of goals and deliverables are critical elements in the success of every project.

Scope Negotiation

Models of project scope identify the functions and features to be delivered by a software project.

In CHARISMATEK's scope models, scope is defined in terms of objective, identifiable and traceable deliverables that have a known relationship with cost and schedule.

Developers and clients utilise this information in negotiating exactly "how much" and "what" functionality can be delivered, given a project's budget and/or schedule. If the budget or schedule is exceeded, functionality can be reworked, removed or flagged as deferred until a satisfactory outcome is achieved.

Alternative strategies can also be explored. For example, functions can be flagged as to their priority in the delivered solution. Increments for a staged delivery can be chosen, whether using traditional development methodologies or an agile approach.

Ongoing Scope Control and Management

Throughout software development or delivery, models of project scope are used to ensure that a project continues to progress in a manner that is consistent with the budget and schedule defined by its business case.

Regularly, at project milestones or after change requests, the model is updated to reflect any changes to a project's functionality or features.

Quantifying the updated model at these times immediately highlights when a project has grown to the extent that it can no longer be expected to be completed on time or within budget.

In such cases, the model is further utilised either to define a new budget and schedule, based upon the increased project scope, or as an aid to negotiating what project functionality needs either to be removed from the project or possibly deferred until a later release so as to re-align the project with its business case.

Benefits

CHARISMATEK's scope services provide software developers and clients with a range of substantial benefits, including:

- ▶ Very early identification of incomplete software requirements;
- ▶ Stakeholders have a single, easily-understood, shared view of a software project's deliverables;
- ▶ Improved project estimates - based upon a confirmed and quantified view of requirements.
- ▶ Straight-forward approach for managing requirements change and understanding its impact on budget and schedule;
- ▶ Early identification of likelihood of project budget and schedule blow-out;
- ▶ Standard techniques applicable across all software types and independent of project life-cycle model or development methodology;
- ▶ Software tool support - CHARISMATEK's Function Point WORKBENCH™ provides the analysis and reporting functions to support all the scoping activities.

Further Information and Contact CHARISMATEK

CHARISMATEK Software Metrics provides a wide range of quantitative analysis services focussed on adding value to organisations both delivering and purchasing software related products and services.

For further information on all CHARISMATEK's services, see the CHARISMATEK website at www.charismatek.com or contact CHARISMATEK on + 61 (0)3 9696 1255 or at info@charismatek.com.