Commentary on the draft DSDM Contract

NOTE: the outline contract is available from the DSDM Consortium website, but you should review the terms of the licence carefully to see the use that may be made of the DSDM contract. The licence (in brief terms) allows use, review and modification, but there are limits on the extent to which it can be disseminated, or a fee charged for such further dissemination.

Obviously, the task of writing a contract that is intended to deal with an almost infinitely wide range of circumstances is daunting. If you add into the equation that the genesis of the principles that guided the drafting was in a working party made up of both suppliers and users, then there is clearly going to be at best, compromise, and at worst, inconsistency.

In fact, the aim of the working party was to state in contractual form the principles that should go into the drafting of a contract for a project that both parties wanted to conduct using DSDM. This form should be "neutral", in the sense that it should allocate to both parties a fair share of the risks and rewards given an acceptance by the parties that DSDM would be used for the project. It is an "outline" contract, in the sense that it states, in a contractual form, what the parties should do, and the contractual (legal) consequences that should follow from performance – or defective performance. It is not a "precedent" contract, in other words, it cannot simply be used as the sole set of terms for any project where DSDM is the sole, or even a main element. There are still many things that parties will want to negotiate, and there are things that a contract should contain that are not included here – obligations of confidentiality, for example, and terms governing the parties' responsibilities and liabilities when working on the other's premises are another example.

Why attempt this task? Modern development methods, loosely described as RAD (Rapid Application Development), of which DSDM is an example, have emerged in the light of greater experience in running IT projects. This experience is based not only on greater insight into IT projects themselves, but also on using the benefits of modern technology.

Looking back thirty or so years, the programming languages available simply did not permit the use of RAD or anything like it. Programming was a major part of all IT projects, requiring absolute precision in the statement of initial requirements. It was also a highly specialised occupation – some thirty or so years ago, it was likely that something like assembler might be used as the programming language, which was inflexible, and only dedicated IT staff could understand it. Development took a long time, and hardly involved the user, leading to long delivery times, and systems that were notorious for not being "user-friendly".

Along came more advanced programming languages and techniques, and with them shorter development times, and greater involvement of users. The programming is now a tiny fraction of the total effort of a typical project. This has led to systems that are potentially easier to use by the end-users themselves, but this is achieved by a far greater involvement at all stages of the user community, and it is they who have to take the lead in specifying their detailed requirements, with the supplier involved in implementing those requirements in a finished system.

A method like DSDM will (according to its exponents) be a project that delivers on time and within budget – but there is "no free lunch". The iterative mode of delivery means that there will be earlier deliveries of useful functionality to the business. The principle that all things are reversible seeks to ensure maximum flexibility, so that the user is able to secure just the right system for its purposes. These promises are achieved by compromises elsewhere. The user must make decisions constantly as to exactly what it wants in terms of its requirements, and the supplier is not taking the lead in dictating to the user what the user will have. Even this statement of responsibilities is a reversal of what many contracts see as the correct roles for the two parties, with the supplier in the driving seat and the user passively receiving what it is given.

This has led to a situation where users seek to hold suppliers to old-fashioned contracts which impose absolute end-to-end obligations on suppliers. This is unfair on suppliers. Equally, where suppliers are starting to use DSDM, they are sometimes prepared to do so only on the basis of a contract that is open-ended time and materials. That is unfair to users. There had to be a middle way, and this outline contract seeks to set out the basis on which a contract can be concluded.

RICHARD STEPHENS

However, there are as many types of project as there are projects and no one form of contract is going to be suitable as-is for every situation. Many projects might be purely DSDM, others might use DSDM for a separate part of the overall project which is otherwise being conducted according to some "classical" or "waterfall" method such as SSADM. All these variations cannot be catered for in this form of contract.

Thus, in all cases where the parties want to use DSDM, or they are planning a project which involves DSDM, it is simply not possible to pull these terms off the shelf and sign them without further thought or discussion. This would be very dangerous. They are a good starting point.

A final comment before turning to the clauses in detail. This contract was drafted with English Law in mind, and while a modern and easy style of English has been attempted and excessive legalisms have been avoided, the interpretation rules and legal principles of English Law have guided it at all stages. Other jurisdictions – including Scotland and Northern Ireland – could well have very different rules, and use of these words might not result in the intended effect in other jurisdictions.

Initial capital letters have been used in this commentary to show where a term defined in the draft DSDM contract has been used.

Clause 1

This introductory clause makes a number of preliminary matters clear. It is intended to enable the Parties to use DSDM in the knowledge that the Contract will allow them the flexibility to adapt and make use of DSDM without their actions falling outside the Contract and potentially leaving them without effective contractual cover.

There are some projects where DSDM is the main method used, but there are potentially as many ways of using DSDM as there are projects. Even this leaves out of account "hybrid" projects, where some other method is used for the majority of the work (perhaps a waterfall method) and DSDM is used for some separate part of the work. In all cases, it is necessary to review the terms of this Contract and use and adapt the wordings to reflect what it is the Parties are actually agreeing to do.

The other point to note about this is that what the Parties Agree actually becomes part of the Contract. Many project personnel may feel that what they do is in some way "technical" and divorced from the "legal" contract. In fact, the decisions they make and the agreements they come to day by day are very much part of the legal framework, and this clause at the beginning of the Contract starts to make this point clear. This is taken up in the next clause on the formation of Agreements.

Clause 2

Preliminary matters about how to exchange documents, authentication and so on are dealt with in clauses 2.1 to 2.3. These are important clauses and the additional documents (protocols) will need to be drawn up by Service Providers and Users if they wish to depart from the "default" provisions contained in this clause. Clause 2.4 makes the point that Agreements within Scope can be reached orally. This might shock many lawyers, since they look to at least written evidence of agreements reached, and the provision stands in stark contrast to "normal" change control provisions, which require change control to be signed off. However, as will be seen, there is in fact little risk of there being a dispute under this clause – provided that the Service Provider (and the User) follow a proper scheme of configuration management.

Clause 2.6 introduces a fundamentally different type of Variation that may occur in the context of a DSDM project. The distinction must accordingly be drawn between Agreements within Scope, and Agreements Varying Scope.

The first type of Agreement (Agreements within Scope) is not really a Variation at all – it is just the sort of Agreement that is made from day to day, between User and Service Provider. These Agreements can often be oral in the context of a DSDM Project, and might relate to "micro-management" of the Project. They do not need to be in Writing to become effective because good practice does not necessarily demand that they should be. Such is the dynamic nature of a DSDM

Project, that it might well be impossible for all such Agreements to be written down, reviewed and signed off before Agreements are actioned, although clearly the more important decisions coming out of Workshops will be formally recorded like that and will await formal Agreement before being actioned (as a matter of practice than necessity).

However, many Agreements will be minuted or recorded by what DSDM refers to as the "scribe" and individual Service Providers will have their own internal procedures and standards for documentation. Clause 2.5 then comes into play and sets out a very important principle. Clause 2.5 allows for minutes of meetings (and similar records) to become records of what was agreed – a useful provision that encourages and rewards good (and normal) DSDM practice in this area. The difficulty is that many inaccurate minutes go unchallenged because of the commercial situation in which one or other Party finds itself. This Contract provides an incentive to the Parties to do so, and the consequences of not challenging meeting minutes (e.g. for fear of offending the other Party – a surprisingly common occurrence) could be serious for a Party. Thus, if the Parties follow what is fairly standard DSDM practice in this area, there is surprisingly little opportunity for dispute about what was Agreed within Scope.

However, there is a second type of Variation that must be dealt with separately. It is sometimes hard to define with exact precision the Scope of a DSDM Project, since the very first statement of requirements for the purposes of the Feasibility Study will be at such a high level that it will be meaningless as a watertight definition of the work to be performed. The statements of Scope agreed in the Business Study are more reliable, but are still at a higher level of detail than is the norm for traditional, waterfall projects. Arguments about the Scope are therefore likely to be difficult to resolve by reference to existing documentation. The requirement of Writing is intended to provide some guarantee of certainty in this area. As long as something is within Scope and can be justified by a Written document, this should assist the Parties to identify the limits of their obligations and entitlements. To assist in this process, there are a number of instances in the Contract where the requirement of Writing is specifically called for. These are not the only instances where Writing is required, and are included only for clarity.

The concluding words of clause 2.6 should be viewed with care. It is in many cases virtually impossible for a Service Provider to provide watertight estimates of the consequences of changes being made now. In any case, in a DSDM Project, the situation is so fluid, that estimates are bound to be only reliable taking into account what is known at the time the estimate is made. Moreover, one of the claimed advantages of DSDM is knowing when a project is non-viable. Since much of the detail of a DSDM Project is decided as the Project proceeds, Service Providers may wish to provide "best estimates" under this wording to provide some protection against their estimates proving incorrect.

However, note that "loose" estimates can only be given where it is "appropriate" and that the Service Provider will gain no protection simply by routinely giving wild or open-ended estimates. In addition, note that the Service Provider must give a reason – and that protection for incorrect estimates is only effective when the estimate is out for the reason given. As will be seen elsewhere, this is an attempt to provide an incentive to greater openness in dealings between the Parties.

Finally, clause 2.7 deals with some important aspects of agency law. It should be of no concern to either Party to verify the authority of each individual from the other Party with whom it has dealings. For one thing, there could be many individuals involved in a DSDM Project, and many of them could be third party or agency staff. In such circumstances, given the fast moving nature of a DSDM Project, it would be unworkable if a Party could not rely on the statements and Agreements made by appropriate members of the other Party's "staff". The exception is for Variations to Scope – these will normally involve considerations beyond what the average user in a workshop could reasonably deal with and so each Party must nominate someone who can provide the requisite authority. Note that this nominated person is not the only person put forward who can Vary the Scope, it is just that the Parties can use this person (or his or her nominee) with safety.

For completeness, it is not the intention that the Authorised Representatives Schedule should contain a list of personnel with details of the limitations on their authority: this would be contrary to the purpose of DSDM. The persons named in that Schedule should in fact have full authority for the purposes of this clause.

Clause 3

See the following sections of the DSDM Manual for a description of what the Feasibility Study is about:

- Section 11.2.1 describes the aims of the Feasibility Study
- Chapter 13 contains a formal description of the products of each phase, including the Feasibility Study

There are other sections dealing with aspects of the Feasibility Study, such as

- Section 17.3, dealing with estimating techniques
- Sections 22.6 to 22.9, dealing with modelling
- Sections 8.3 and 23.1, dealing with facilitated workshops
- Appendix B, dealing with test planning

The key point to emerge from such sections is that a variety of processes can be applied to come to a successful conclusion of the Feasibility Study stage. The application of such techniques will be by Agreement between the Parties, but all will depend on the precise circumstances, each having slightly different outputs and attendant risks. As elsewhere in any development method, the application of the particular processes or techniques lies in the realm of art, not science.

An important point to note about this clause is that there are alternative versions. It is possible that the Parties will work together on the Feasibility Study, in which case version one will be appropriate. The more usual scenario is that the User will first complete the Feasibility Study on its own, perhaps with third party consultant support, and the Project (in the sense of the Parties working together) will commence with the Business Study and build on the basis of the work done during the Feasibility Study. For this, version two is appropriate.

A very important point about the first version arises if the Parties Agree to abandon the use of DSDM. This effectively means the cancellation of the whole Contract and a new contract must be entered into. If the Parties simply carry on regardless, they will find that the work is not covered by any written contract at all, since the DSDM contract will have terminated automatically. This is an important point for the Parties to remember.

If version two is the one chosen, the User must very carefully note the effect of clause 3.2. Some Service Providers may perform something approaching an independent verification, but this is not automatic, and it should not excuse the User from the responsibility for standing behind the work that went into its (the User's) Feasibility Study.

Clause 4

The Business Study is the second of the two important preliminary stages of DSDM. It follows on from the Feasibility Study and builds on it, so to speak, and becomes the fundamental basis for the rest of the project. The following sections of the DSDM Manual deal with the Business Study:

- Section 3.1.2
- Section 11.2.2 describes the aims of the Business Study
- Section 11.3 describes the interplay between the Business Study and maintainability
- Section 13 gives a formal description of the products of each phase, including the Business Study
- Section 21.3 stresses the need to develop an approach to testing as part of the output of the Business Study – this includes providing an definition of the "fitness for business purpose"

Techniques relevant to conducting the Business Study are also described at the following sections:

- Section 17.3 deals with estimating techniques
- Section 20.4 deals with the need to decide on configuration management before leaving the Business Study

- Section 21 is also relevant, as it deals with testing, the basis of much of which is determined during the Business Study
- Sections 8.3 and 23.1 dealing with the use of facilitated workshops
- Appendix B, dealing with test planning

The Business Study in a sense "finesses" the Feasibility Study, and takes it the logical stage further. As well as considering again if the project is susceptible to DSDM at all, it looks in more detail at the question of scope, defining the deliverables (including the minimum usable subset – or "MUS" – of functionality). The Business Study also looks at non-functional requirements, such as requirements for maintainability, and defines with some precision many of the issues that will go into defining what is meant by "Fit for Business Purpose".

The important thing about this stage is that DSDM is less "binary" than traditional or waterfall methods. DSDM does not define absolute sets of functional and non-functional requirements (apart from the MUS) but rather looks to the parties to agree prioritisation schemes for both functional and non-functional requirements. In particular, the User is involved in the frontline of deciding schemes for prioritising its requirements (functional and otherwise).

It looks also – and this is crucial for the rest of the Project – at the resources required for the successful accomplishment of the Project. These resources are both human (e.g. user personnel) and technical (e.g. the System Architecture Definition). The statement of such resources sets the limits for what can be achieved in the rest of the Project, and the Parties need to consider carefully what things should be set out as specific Dependencies in the Dependencies Schedule.

Another important point about the Business Study is that it is not a standalone product, capable of itself of defining either the content of the Project or the methods agreed to achieve it. The Business Study "feeds into" the rest of the Project, and it is during the rest of the Project (the Functional Model Iteration and the Design and Build Iteration) that the products of the Business Study are used as the basis for further work of refinement and then are worked up for Implementation.

The Business Study also provides the standards (as initially stated) by which the remainder of the Project will be judged, including many of the details of what will be regarded as "Fit For Business Purpose".

Clause 5

Together with clause 6 below, this is the heart of the contract, dealing as it does with Development. Development is very different from more traditional project methods and, in the words of Principle 5 of DSDM, "iterative and incremental development is necessary to converge on an accurate business solution". This expresses the way DSDM proceeds to deliver an appropriate solution against the User's business needs.

The difficulty in expressing this in contractual terms is immense. In essence, all a written contract can do is provide a framework for the myriad of decisions (Agreements) to be made and to deal with the commonest problems that will arise. Thus, the first point to note is that a contract for DSDM does not, like many other contracts for systems, and cannot, set out in detail in advance all the delivery dates for a detailed lists of deliverables, or even a detailed description of what finally will be delivered and Accepted.

The second point follows directly on from the first. The actual decisions that matter, the day-to-day management of the process, decisions about the solution required, resourcing, and the rest of it, will be decided by the Parties, often in face-to-face meetings, often by email. Decisions may be made with a User who is, in management terms, relatively lowly in the structure. Agency and authorisation principles (who is authorised to do what) are, of course, dealt with in clause 2.7 above, but the important realisation is that authority will have to be delegated down, sometimes to levels that organisations are unused to allowing. This itself will have tremendous impact on User organisations.

This has the effect that the contract (in the general sense, what the Parties have legally agreed and continue to agree from day to day) is not made of documents that will necessarily be scrutinised in advance by lawyers and commercial people, but will be made by those "at the coal face". While this

may send shudders of shock down the spines of many lawyers, this is simply the way that DSDM works: there is no sharp dividing line between "technical" and "legal". Indeed, much of modern systems development already proceeds (or seeks to proceed) in this way.

This Contract tries to deal with this and the consequences of operating in this environment. It has therefore to deal with the likely problems that will fall out of this mode of working. It raises profound questions for the internal organisation of businesses, both service providers and users, since they must both grapple with the consequences of allowing relatively junior personnel to "run the show". Some will want to institute more specialised training for such staff, and there is the prospect of closer supervision by Service Providers of their project staff. This will certainly be important in such matters as Agreements as to Varying Scope.

Clause 5.2 deals with the details of what the Parties have to Agree following on from the Business Study. It is important to note that this is Agreeing the detail of the framework that should already have been Agreed for the Business Study. The Parties will need to ensure that at this stage the Scope is adhered to and Written Agreements made where there is any Variation to Scope. There may be need to have recourse to the dispute or disagreement resolution provisions at some point, since there is a natural tension between Service Provider and User, with the User often pressing to increase the Scope and the Service Provider concerned to keep the Project within the manageable bounds of the Scope.

Clause 5.3 describes how it is these various Agreements will be reached. The basic division of labour between the Parties is that the User should be deciding the ultimate "shape" of the solution (functional and non-functional requirements) by applying the MoSCoW Rules and the Service Provider providing the management and technical resources to achieve this while advising the User on how to proceed. The Parties, at a practical level, have to take care that they do not cross over into the other's territory. The Service Provider is not designing the System for the User, and the User is not telling the Service Provider how to go about doing it. This clause is intended to describe the correct balance.

While Testing and Acceptance are dealt with in clause 6, the Parties will, as a matter of practice, be Agreeing details of Testing as they proceed through the development stage. It is important to provide the criteria which are the basis of the Tests Agreed. The sole criterion, according to DSDM theory, is Fitness for Business Purpose. This is perhaps, from a lawyer's perspective, an unfortunate expression, since the meaning for a DSDM practitioner (and perhaps more widely in systems development) is very different from what a lawyer would understand by the term.

A lawyer would probably understand by "fitness for purpose" that the system to be delivered meets all the purposes as made known to the supplier by the user prior to contract. DSDM takes the converse approach. A system is "fit for business purpose" when it meets the requirements of practical, everyday use. Management jargon would say this is "validation of requirements, not verification against specifications", but the crucial difference is that the law looks backwards, towards compliance with a previous agreement or understanding, whereas DSDM looks forwards, towards how the System is to be used in practice.

This Contract expresses this by clauses 5.5 and 5.6. It is vital that the Parties understand their implications. One of the chief purposes of DSDM is to give to the User the System it needs and not necessarily the system it thought it wanted. Many Users at the outset of projects have fixed and firm ideas about what system they think they need. Long specifications are written with detail about what is "indispensable". DSDM attempts to cut through this and expose the core of what is really essential and to provide a System that will meet needs within the context of a Project that will not overrun its dates (or budgets).

Again, it is to be anticipated that the Parties may well have recourse to the dispute and disagreement resolution procedure at clause 12 for controversial decisions in this area.

Clause 6

This clause with clause 5 represents the key provisions relating to the development stage of the Project. Acceptance is important and the legal consequences are set out explicitly. If something is

Accepted, the legal consequence is that the work or the product Accepted cannot be rejected because a defect is later found (see clause 6.6.1).

The benefit to the Service Provider (and to the Project) is the certainty that such Acceptance provides. The User still has a remedy in the event that the defects subsequently show up: a claim in respect of those defects (clause 6.6.1). It is likely that the Service Provider would choose to re-perform the relevant part of the Project. The other point to note is that the nature of DSDM is such that an Accepted part may go into live operation as part of the User's business. This has knock-on effects in terms of the Service Provider's liability provisions – there is a very real possibility that something might be put into use, that is defective, and which causes the User loss. Service Providers will want to include this sort of event in any exclusions or limitations of liability.

It is the User's duty to Test such matters as will satisfy it that the things being Tested really are fit to be Accepted. This is made clear by clause 6.2 which limits the Service Provider's obligation to advising on what the User should do in this respect. A further point to note is that, as the Project proceeds, the User will need to factor in Tests that verify that the System as a whole works together (regression testing), as well as testing that the part being Tested is Acceptable.

DSDM is a method that is incremental, and backtracking is part of the fundamental nature of DSDM projects. A subsequent Agreement to Vary something already Accepted is therefore specifically saved by clause 6.6.3, so Acceptance does not mean the part Accepted has to remain in place for the rest of the Project.

What is legally more contentious is where Tests are failed. Clauses 6.7 to 6.10 deal with the consequences of failure, and are lengthy and, of necessity, somewhat detailed. The basic principles are as follows.

1. The first question to ask is whether the failed Tests (or any part of them) were failed as a result of the Service Provider's failure to discharge its obligation of reasonable care and skill. Two different regimes apply in each of these separate cases. While the User is entitled to the MUS, it is not necessarily the case that every failure of a non-MUS deliverable means that it was the Service Provider's "fault". For example, if the MUS in a particular Timebox was estimated at 75% of the effort, but in fact took 95% of the effort, then it may well be the case that the remaining work scheduled for that Timebox will not pass Tests. However, if the reason for this is that the User simply decided to reallocate the effort in the Timebox to the MUS and away from non-MUS matters, then this is not something for which the Service Provider is responsible.

2. The User is entitled to the MUS, so

    a) if the reason for not providing the MUS is as a result of a defect or omission in the Service Provider's work, the Service Provider must in the first instance attempt to redress the failing and provide the MUS. If the Service Provider cannot do this, the Tests are failed. The Service Provider must also make up for the time lost by providing an additional and equivalent amount of time to the User for other work.

    b) if the reason for not providing the MUS is not as a result of the Service Provider's performance, then the User can still call for the MUS, but in this case must pay for all further work done in providing that element of the MUS, and may therefore well have to make decisions applying the MoSCoW Rules in re-prioritising other work, possibly de-scoping the Project in other areas.

3. As regards aspects of the Project other than the MUS

    a) if they are not completed or fail for reasons resulting from a defect or omission in the Service Provider's performance, then the Service Provider must make them good and must make up for the time lost by providing an additional amount of time to the User for other work in other Timeboxes.

b) if they are not completed or fail for reasons not resulting from a defect or omission in the Service Provider's performance, then the matter ends there, and the Parties can always Agree to try to implement them in some other Timebox.

4. The consequences of ultimate failure are grave, and the User could even terminate the Contract for breach of contract. As has been seen, such failure could only happen if the Service Provider failed to provide a part of the MUS as a result of a defect or omission in its work, and then failed to put that right before a repetition of the relevant Tests.

5. If the User Accepts the whole of the System but without a part of the MUS that failed, then it would receive a price reduction, but could still sue for any loss provable as a result of the non-delivery of the failed part of the MUS.

Clause 7

Charging and times of payment will depend, perhaps more than other parts of the Contract, on what the Parties negotiate prior to entering into the Contract. The presumption here is that T&M will be the norm, but that is not inevitable. However, it should be pointed out that the Contract in its form as presented here puts onerous obligations on the Service Provider effectively to put the User's interests first, and there is a potential conflict between this and fixed price (where the Service Provider will have the incentive to maximise its profit margin). If fixed price is chosen, the Service Provider may well want to revisit some of those other obligations.

The alternative would be to agree some cap on the T&M – a maximum figure that cannot be exceeded. Again, this is of benefit to the User in knowing the budget, but is risky to the Service Provider in the event, for example, that the User is undisciplined in applying the MoSCoW Rules. The inevitable question then would be, who takes the risk of the cap being exceeded – the Parties would have to provide for this eventuality in a change to the standard form.

A further and very important point is that the Parties may want to break the Project into two: the first part consisting of the part up to the end of the Business Study, and the second consisting of the part following on from that. It is perfectly conceivable that the first of these two parts could be fixed price, as the effort is more capable of being capped, while the second is T&M. The Service Provider may be required to provide a forward estimate at the end of the Business Study stage for the remaining development work. Indeed, many Service Providers may see this as their preferred option. The Parties would need to specify what the consequences would be for exceeding the budget in this case.

There are advantages in this two-stage approach for the User too: given that cancellation may be the result of the Business Study, a fixed price for this stage does cap the possible loss and make budgeting easier.

In reality, the Parties will probably see this as a major part of what they want to negotiate, and the outline contract should not be seen as setting out any principles in stone. However, the important point is that it is also the case that Parties may well want to make changes elsewhere in the contract depending on what changes are agreed to payment terms.

Clause 8

Clause 8 is crucial from the legal perspective. The Contract at many points defines what it is the Parties have to do, and their respective roles and obligations. This has to find a legal expression and that place is these clauses. The Parties must carefully understand them and their ramifications. They answer the question, "by what legal standard are the Parties' legal obligations to be judged?"

It should also be noted that this clause, together with all the other more specific provisions of the Contract replace what in other contracts might be gathered together under a clause entitled "warranties" or some such.

The expression "reasonable care and skill" is one found in section 13 of the Supply of Goods and Services Act 1982. Obviously the general law will apply to judging the performance of the Parties. What is "reasonable" in this context will depend on a wide variety of factors, including the terms of the

Contract as agreed and all the Agreements reached by the Parties, as well as looking at the wider situation.

Sub-clauses 8.1.1 and onwards provide specific obligations that apply to the Service Provider. Clause 8.1.4 is likely to be controversial and deserves a comment of its own. This requires the Service Provider to report to the User its own actual or apprehended failings. So, a Service Provider must provide information on a timely and proactive basis about its own failings, such as bugs in software that are proving difficult to eradicate. It is part and parcel of the cooperative nature of DSDM.

Clause 9

All IT projects will depend, to some extent or another, on the cooperation of the Parties. This clause, which is broadly drafted, is intended to reflect the duties of both parties in working together to achieve a result. The important point to note about this clause is that it simply tries to deal with the consequences of Dependency Failure, and does not try to allocate "fault" or "blame" to either Party.

A preliminary point is that many exact Dependencies should be stated by the Parties and included in the Dependencies Schedule. This will often be part of the Feasibility Study and Business Study, which will, for example, name roles of individuals that are essential to the successful performance of the Project. Other Dependencies will be Agreed as part of the Development without their necessarily being defined as formal "Dependencies". They are all Dependencies as defined, however, and the consequences of any of them failing is described in this clause.

The basic principle is that the Parties must cooperate in the event of a Dependency Failure, but the User's failure to cooperate in some essential way does not alleviate the Service Provider's duty to continue with the Project as much as it reasonably can. What happens then is that the Party which has been held up by the Dependency Failure is excused to the extent it cannot continue because of the Failure. It is important to note that Dependency Failure is not a general excuse for suspending the Project – on the contrary, everything must be done to work around the Failure as much as reasonably possible. See also clause 9.5 on charging for the effects on the sums payable by the User caused by working around a Dependency Failure resulting from a matter within its control.

There is an exception to this. It may be that the Service Provider will want to claim the right to walk away because of the extent of the disruption caused by the Dependency Failure. This situation is handled by clause 10.2 under the heading of cancellation. Clearly, a Service Provider will not want to keep staff sitting around idle for too long a time, even if the costs of the disruption are being paid for by the User. In that case, a right to cancel is provided to the Service Provider.

Clause 10

The Service Provider in fact has limited rights to cancel. Specific rights to cancel exist at the Feasibility and Business Study stages. The Service Provider can recommend cancellation, and this is potentially an important power. The reason for limiting the Service Provider's right to cancel is so that the User is not left in the middle of the Project without any means of continuing with that Service Provider. The exception to this is where the Project has become so affected by Dependency Failure that the Service Provider ought to be excused from continuing. This might happen if, for example, the User suffered a fire at the Project site and the only alternative was for the Project to be put on ice for a year.

However, the Service Provider's power to recommend cancellation is still an important one. Doubtless, a Service Provider will not want to exercise it too much or the wrong impression will be given to the User, but at the appropriate point, it could be an important way of the Service Provider highlighting particular problems. It has the advantage for the Service Provider in that it can escape liability for the Project going wrong subsequently for reasons contained in the recommendation. The fact that independent reasons have to be given is a guarantee that the Service Provider will not exercise this right frivolously throughout the Project so as to try to gain some advantage of excluding liability for the outcome of the Project.

This is different from the User's position, since a User can cancel at any time. Cancellation is not a statement about either Party's performance on the Project, and there will usually be sound business or technical reasons for cancelling a Project. One of the claimed advantages of a method like DSDM is that it provides early warning where a Project is not going to produce the results that were to be hoped for (as perhaps set out in the Feasibility Study). Where a Party wants to terminate for breach or in some way that leaves open a claim for damages, then the termination rights under the Contract must be used (not included in this outline Contract).

Cancellation is intended to recompense the Service Provider for all work that has been done and for all commitments entered into that cannot be undone (e.g. fixed contracts for third party staff). Reasonableness should not be a prime consideration. The User may request a cancellation statement from the Service Provider in order to check the costs in advance of cancelling.

However, there is no doubt that some Service Providers may find the cancellation somewhat burdensome: if the cancellation occurs early, the Service Provider may not have had the opportunity to earn sufficient money to make the Project profitable (and to cover any pre-contractual expenses associated with the bid costs). It may well be that the area of negotiation here will be a sliding scale of compensation, with more paid for termination in the early stages, decreasing towards the end of the project.

Clause 11

This clause could well prove controversial, but it reflects what is perhaps the beginning of a trend in providing more generous source code rights to users. It is likely that the Licensing Schedule could be quite a complex document, taking account of individual third party owned materials, and the specific licensing relevant to them.

The basic rule is that the draft contract does not alter ownership rights. All the user gets is a licence. The big difference is between Foreground and Background IPR, Background being what each Party starts with, Foreground being what a Party develops in the course of the Project. If it is Foreground, the user is entitled to source code, subject to the rules described in clause 11.5. The crucial part is that this right extends to allowing third parties to work on that source code for further development (clause 11.6). Again, this may be controversial for some Service Providers, who may wish to prevent certain specified competitors from working on source code that they had developed. This may be written into the Licensing Schedule, but there may be competition law implications in so doing.

Clause 12

A dispute resolution clause is essential in a contract of this nature, and the provisions in this case are quite elaborate. A basic distinction is made between a dispute and a disagreement. A dispute is basically any sort of dispute about the course of the project or the interpretation of the Contract. A disagreement is where the Parties cannot come to an Agreement where they are required to do so.

Different regimes are given in different Schedules to be followed in each of these cases. It may be that the Parties want to include some sort of adjudication provision in the (presently blank) Adjudication Schedule, a clause that would allow an expert to provide a speedy and binding determination of issues as they arise in the Project (perhaps akin to the adjudication schemes used in the construction industry).

The possible conclusion of the two different regimes is also given. Disagreement might possibly lead to cancellation, but not before a notice has been given that this right is to be exercised, thus allowing further time for the Parties to come to an Agreement. Again, this is potentially controversial, as a Party could always fight to the bitter end and then exercise its right to cancel.

Finally …

These clauses would not be the whole of a DSDM contract – there are many other provisions that individual parties would want to include dealing with many specific points. Equally, there may be projects where some of the clauses in the draft would not be appropriate. Clauses on termination, confidentiality, liabilities are all obvious candidates. In fact, when the obligations are looked at in

more detail, it seems that these (and other) clauses would benefit from being looked at afresh and some re-drafting done for the benefit of both Service Provider and User.

In some senses, legal drafting might well find different avenues for expression – the Licensing Schedule is an obvious example, and also drafting up the Dependencies Schedule to deal with specific risks (such as the risk that a particular core software product proves to be inappropriate). It is areas like these that need to come under the legal spotlight, as the areas of risk in a DSDM project shown up by this outline contract come to be fully appreciated.

It is important that the outline should receive as many comments as possible, so do please let me have comments – on the principles as well as on any drafting mistakes.

Richard Stephens
richard.stephens@masons.com