Software Sector SR&ED Workshop

Background and Highlights

Key Concepts, Tests and Challenges

The Software Guidance www.cata.ca/sred-guidance/

- Developed by the community with the assistance of the CCRA as clarification of Information Circular IC 97-1.
- The CCRA agrees with the concepts and that the guidance respects the legislation.
- The Guidance is the base for joint training of the CCRA's staff and the community.
- Draws from the legislation, 3 basic criteria and the concepts of IC 86-4 R3.
- Clarifies problematic interpretive issues associated with IC 97-1.
- Provides extensive guidance on documentation useful as evidence and on how to document a claim.

The Law Supports:

Key Tests

- ***Scientific Research and Experimental Development (SR&ED),*** namely, activities involving
  - *a **systematic investigation or search*** carried out in a field of science or technology
  - *by means of **experiment or analysis….***
- ***"Basic and Applied Research"***
  - *for the "**advancement of scientific knowledge**"*
- with or without a specified practical application in view.
- ***"Experimental Development",*** namely, work aimed at
  - "***achieving technological advancement"*** for the
- purposes of
  - ***creating "new or improved devices, products or processes".***

- the engineering, design, operations research, mathematical analysis, computer programming, data collection and testing needed to support the basic or applied research or experimental development effort,
  - *but...only that which is truly needed.*

The Law Does Not Support:

Key Tests

- Solely routine design development, routine engineering development,

- Commercial production or commercial use of the devices, products or processes developed through SR&ED.

- Training and learning, unless it is related to the needs of the research or experimental development,
- SR&ED conducted *outside Canada*,
- Research in the Social Sciences or Humanities,
- Market research.

Key Tests: *the 3 Criteria and the Law*

- **The advancement of *scientific knowledge* arises directly from the legislation without caveats,**

- **The advancement of *technological knowledge* is only an indirect test implicitly tied to the application setting by the legislation and its *technological advancement test*,**

- ***Technological Uncertainties and advancements* are intrinsically linked, in contrast to other technical uncertainties.**

Some Definitions

- **Computer Science** is where advances in hardware and software arise from the unique application of combinations of technologies and ideas to developing methods of manipulating, structuring, or communicating information or data through computers.
  - It involves theoretical and applied studies of the development and use of computers for information (data) storage and processing, mathematics, logic, etc.
- **Information Technology** is the body of technological knowledge associated with collecting, storing, manipulating and communicating information using computers and communications systems (IC-97-1).

What is Important in CS-IT

- It is **how** the data is manipulated, organized or communicated, and **whether** the methodology represents an advance over the commonly available technology of data or information manipulation or communication; and
- The 3 Criteria of IC 86-4 R3

Key Concepts from the Guidance

- Determining that a technique, solution, construct, methodology, etc. **will or will not work** can constitute an advancement.
- **Advancements can** occur in our knowledge of new techniques, but also in our understanding of **how** best to use them and **which** solutions, techniques, etc. are appropriate for which purposes.
- True technological advancements would be recognized by other developers as significant, important and useful knowledge (the peer test).

- **Simply learning on the job is not an advancement in the sense of the 3 criteria.**

- Not all technologically based uncertainties requiring experimentation are true technological uncertainties.
- True Technological Uncertainties are associated with technological challenges, and limitations in the existing technology base.
- True Technological Uncertainties require finding new methodologies, techniques, solutions, constructs, etc. to overcome them.

- True Technological Uncertainties can always be associated with a Scientific or Technological Advance.
- The **process** or **discipline** inherent in software development should normally be the key to demonstrating the existence of a systematic process of experimentation or analysis.
  - **This must be followed and documented.**

Challenges with Software Claims

- **The law focuses on how the Technology Base is changing and being improved.**
  - This happens in a matter of months; what was new is often established practice when a claim is filed or assessed.
- **The law Focuses on Evidence**
  - Knowledge of the problems and what was done moves on with the developers - ***some sort of record is needed.***
  - Records of the development process are often not properly archived and, only poorly maintained, if at all - ***but a record is needed.***
- Constraints and technological challenges are often due to the specific nature of the product, which technologies are to be used, and/or the setting in which the software is to be used.
  - ***Often only company's developers can understand or explain this and documentation is needed.***

- **The law Focuses on Evidence**
  - Both components developed through established practice and those that require *Experimental Development* are often found in the final product.
    - Only work **Commensurate With** and **Directly Related** work can be included in the claim.
    - An allocation system is needed.
  - All low level software development activities look like the application of standard

practices

- ***Evidence is needed that low level activities are appropriately being assigned to the ED project.***

Challenges in Reviewing Software Claims:

- Focus on understanding the technologies, what they could or could not do, the constraints, the challenges, the management processes used to find and prove out solutions; and
- Focus on understanding the systems, approaches or internal controls used by a firm to put together and validate its claim.
- Identify the types of natural evidence that are normally generated by the firm given their management approach and start here.
- If possible, find out what happens to this evidence.

Summary of Ideas Emphasized

- Advance is from a stated base.
- Business environment modifies "generally available" re talent and knowledge.
- Delineator of eligibility is found in Advance and Uncertainty <u>not</u> in experiment or analysis.
- Advancement is as seen by a reasonable group of peers.
- Evidence of eligibility can be found in contemporaneous documents generated naturally.
- Claimant must structure the project correctly.
- Science reviewer should start by probing for eligibility rather than start by teaching criteria.

<u>Issue of the Business Environment</u>
What Does the Business Environment Modify?
<u>In the Business Environment</u>

- S/T Uncertainty arises when solution is not readily apparent to **appropriately skilled and experienced personnel**
- Uncertainty arising from **lack of appropriate expertise** or **failure to use commonly available information** does not constitute an S/T Uncertainty

<u>SR&ED not a Training Incentive</u>

- However:
  - learning arises on eligible projects
  - information available in one business environment may not be available in another
  - the most qualified individuals for a particular task may not be reasonably available to some business environments

Assessing in the Environment

- Claimant expected to **hire** "appropriately skilled" as would a competitor of same size
- Claimant must demonstrate reasonable efforts to **search for existing solutions**
- 3rd party work is to be evaluated in 3rd party environment
- Some claimants may take technical risks that others would avoid

Core Technologies
Rich vs. Not Rich in Candidates for Eligible SR&ED Projects?
What Would I Expect?

- Rich in Candidates
  - initial releases of saleable software
  - software developed for core of claimant's business
- **NOT** Rich in Candidates
  - subsequent releases of saleable software
  - managerial, reporting, efficiency and support software (incl. financial)

HOWEVER

- **ANY project which can meet the three criteria IS ELIGIBLE regardless of the business environment or its intended business objective.**

Rich in Candidates

- initial releases of leading edge saleable software
  - often gains market advantage based in its technology over existing software
- software developed as part of the technological foundation of claimant's business
  - there is often a technical constraint to overcome that explains why this software has not been developed earlier

NOT Rich in Candidates

- Feature extensions of saleable software
  - Often same architecture and tools or simple adaptations - no S/T Advancement
- managerial, reporting, efficiency and support software (incl. financial)
  - Often coding known algorithms with known tools or adaptations or customization of established systems

The Message

- For areas rich in candidates
  - you can probably identify an S/T Advance and Uncertainties - LOOK FOR THEM
- For areas **NOT** rich in candidates
  - for most projects there is no S/T Advance and the Uncertainties are routine.
  - be sure that you can identify an S/T Advance sought and S/T Uncertainty before developing a claim.

Constraints – Technology & Environments

First Principles

- Laws of nature define the environment which create the constraints within the area of physical sciences,
- Similarly a computing environment provides the constraining boundaries for software/ computing sciences.

Constraints => Technological Uncertainty

- Constraint is a foundational element of technological uncertainty (TU),
- Absence of any constraint => Absence of TU,
- Resolution of a TU (constraint) =>TA.

Technological uncertainty - "First Principles"

- Technological uncertainties are fundamentally derived from :
  - Competing constraints,
  - Non deterministic environments,
  - Lack of knowledge required to determine the characteristics of a given environment.

Typical Sources of Constraints

- Environment
  - Architecture
  - Size of system
  - Footprint
  - Scaleablility
  - Inter-operability standards
  - Resources & services
- Requirements
  - Cost
  - Functionality
  - Response
  - Throughput
  - Concurrency
  - Stability
  - MTBF
  - Legacy & Compatibility Issues

The Issue Sheets
S/T Advancement

- S/T Advancement
  - discovery of technical knowledge that advances the understanding of scientific relations or technologies from the perspective of the claimant's business environment,
  - **applicable beyond the specific situation in which the discovery was made,**
  - **advances from a <u>defined base</u> of knowledge**
  - **strong evidence of a change and improvement over what could be done with or achieved with the base demonstrates S/T Advancement.**
- S/T Advancement

- brings new knowledge to claimant which had **not** been generally available to him in the base when he started the project

- Not all new software comes from Advancement in Information Technology,
- Some new software is simply a new commercial product produced without S/T Advancement.
- There is no S/T Advancement if software:
  - was designed and coded using well known tools and constructs in known combinations,
  - provided no additions to the claimant's generally available knowledge base.
- Evidence of base for Advance
  - credible third party statement of capabilities of technology,
  - description of technological limitations of available product or technologies,
  - claimant's own prior product or test results,
  - **simply claiming development of new features or functionality does not satisfy the test for Advancement.**
- In making a claim the claimant should:
  - consider whether project is in IT/Computer Science or in some other field,
  - define technological success (quantitatively if possible),
  - define the technological base from which the Advance was sought.
- Questions useful in identifying Advancement:
  - what technique has not been used before?
  - how does this compare with earlier solutions?
  - what earlier technical constraint has been overcome?
- Questions useful in identifying Advance (continued):
  - what would a peer consider to be a technological advancement?
  - for what technical reason was this not done earlier?

Experiments in Software SR&ED

- SR&ED Experiments :
  - design test conditions specifically to resolve a scientific or technological uncertainty,
  - make observations that contribute to filling gaps in technical knowledge.
- SR&ED Experiments (continued):
  - **new knowledge is applicable beyond the system under test,**
  - Results will be systematically recorded for almost any SR&ED experiment. **Claimant should retain these results.**

- Examples of SR&ED Experiments
  - Assessing performance of prototype code written to resolve design alternatives,
  - Pre-designed probes to assess unspecified properties of software systems.

Experiments not within the Context of Software SR&ED

- series of probes often not sequenced in a pre-plan,
- when objective is to solve a very immediate problem of little general utility,
- when result is not usually generalizable.

Experiments not within the Context of Software SR&ED

- Examples of experiments without S/T Advancement:
  - setting parameters by ramping up or ramping down a setting or even by testing performance against some defined set of combinations,
  - Debugging (finding coding errors or even resolving misunderstandings about specifications).

Fixes, Feature Extensions and Testing

- Key Points related to these activities:
  - occur both before and after commercial release. Timing does not determine eligibility.
  - a project is complete when it has been shown that the Technological Advancement is consistently demonstrable within the constraints defined,
  - may or may not be included in a SR&ED project, but normally are not stand alone SR&ED projects.
- Key Points related to these activities:
  - when essential to the consistent demonstration of the Advance, they are eligible,
  - when commensurate with the need to provide an environment to prove consistent demonstration of the Advance, they are eligible.
- Key Points related to these activities:
  - always eligible as support activity when commensurate with the needs of an otherwise eligible project,
  - **ineligible** when the primary purpose for testing is
    - user acceptance,
    - marketability,
    - competitive assessment.
- Key Points related to these activities:
  - when claimed as a separate project, as is often the case with subsequent software releases, the project must meet the 3 criteria **and often does not do so in practice.**
- Record Keeping
  - software development often contains a mix of eligible and ineligible activities - especially after first “in use”,

- to support a claim, claimant should be able to show that he has recognized ineligible activity - especially following first “in use”.

System Uncertainty

- System Uncertainty arises with:
  - unexpected interactions between components,
  - inadequate information about components.
- A System Uncertainty can be a reasonable S/T Uncertainty,
- System Uncertainty per se is **not** sufficient for eligibility - the 3 criteria must be met.
- System Uncertainty is not a reasonable basis for a claim if activity is limited to:
  - learning from a vendor,
  - optimization,
  - solely routine design, integration and testing challenges,
  - integrating components which were designed to work together and generally did so in the claimant’s experience.
- When claiming work with a vendor, claimant activity cannot be confined to making complaints. Rather the claimant must **identify fundamental problems** (not just bugs) and participate in the search for a solution (not just verifying a new version for the vendor).
- In making a claim, the claimant should:
  - provide clear evidence that there was a system uncertainty,
  - show that the system integration could not be achieved through standard development practices (should describe alternative approaches tested),
  - additionally claim a specific S/T Advance (from a defined base) in IT/Computer Science.
- The scope:
  - Only work commensurate with and directly related to resolving the System Uncertainty can be claimed in the name of “System Uncertainty”.

Interfaces to other Packages and Users

- The Situation:
  - frequently impossible to show an S/T Advance because the approach to solution is either well known, a simple adaptation or not applicable beyond the present project.
- The Situation (continued):
  - frequently impossible to initially define S/T Uncertainties because it is not clear that the first proposed approach might fail,
  - uncertainty as to how best to display information to users is not an S/T Uncertainty in the field of IT/Computer Science.
- With packages, valid claim will probably be based upon:
  - new constructs rather than simple adaptations,
  - serious alternatives having to be explored through analysis or prototype build and

test,
  - development of new knowledge that is applicable beyond the present project.
- For GUI and other User Interfaces:
  - most of this work is in the **social science fields** and is thus not eligible,
  - if a claim is made in the field of IT/Computer Science the S/T Advancements and the S/T Uncertainties must be in IT/Computer Science.

When is Quality Control Eligible?

- Note that:
  - the organizational department in which activity occurs is irrelevant to eligibility,
  - any activity commensurate with the need to show that an S/T Advance is consistently demonstrable is eligible and this can include the work of the Quality Control Group.
- Ineligible activities:
  - regression testing,
  - testing of a software release that is not itself otherwise the subject of a claim.
- Testing involving a mix of SR&ED/non-SR&ED components:
  - Claimant should be prepared to show that testing activity is not being claimed beyond that commensurate and directly related to proving an S/T Advance in the SR&ED component.

When is Analysis Eligible?

- Note:
  - only rarely is a purely analytical project claimed,
  - analysis is usually eligible as a supporting activity within an otherwise eligible project,
  - analysis involved in planning of a project when that project is not actually undertaken does not qualify the project.
- Eligible Analytic Projects:
  - known methods and tools **can** be employed
  - the analysis seeks an S/T Advanc (which is applicable beyond the present project),
  - meets the 3 criteria.
- Ineligible Analytic Projects:
  - known methods of making a selection between known alternatives,
  - analysis involved in planning of a project when that project is not actually undertaken.

Improving the Process of Software Development

- Note:
  - claims for development of new tools are often eligible,
  - claims for the development of “development processes” (in which tools may be

embedded) are rare, but they can be eligible.

- Development of Development Processes
  - must satisfy the 3 criteria
    - in particular, multiple iterations with testing and refinement **in the development of the process** will be needed to satisfy the criterion of Technical Content.
- must satisfy the 3 criteria (continued)
  - claims for the development of "development processes" based upon a single incidence of an otherwise ineligible software development in a new way fail the criterion of S/T Content.

Tests for Eligibility
S/T
Advancement and Uncertainty

S/T Advancement

- Any S/T Advancement sought or achieved:
  - starts from a declared technological base of knowledge,
  - seeks an achievement or demonstration within constraints,
  - produces a recorded result (which may be negative).

Looking for the Advancement

- Would peers have some respect for the advance sought?
- Can the new knowledge be **applied beyond the present system** specifics?
- Would the claimant need more than engineering + acceptance testing to move from his technological base to his sought advance?
- Is there a technological reason why the advance was not sought earlier?

Looking for the Uncertainty

- Any serious risk that project objectives could not at least be approximated within the technological constraints?
- Were there serious alternatives to the software structure that was eventually chosen?
- Was testing or serious analysis done to evaluate alternatives?
- Was any prototype code abandoned?

Examinations of Eligibility

Focused Tests for Eligibility

- Constraints,
- Knowledge gained,
- Starting / Ending point for technology,
- Objectives,

- Unforeseen obstacles/deviations from plan,
- Peer review,
- Content – (work product).

TU Constraints revisited

- Environment
  - Architecture
  - Size of system
  - Footprint
  - Scaleablility
  - Inter-operability standards
  - Resources & services
- Requirements
  - Cost
  - Functionality
  - Response
  - Throughput
  - Concurrency
  - Stability
  - MTBF
  - Legacy & Compatibility Issues

What is the current state of Firm's technology?

- Is this project a continuation of previous work of the firm?
- In technological terms describe what the firm's technology is currently capable of prior to this SR&ED activity.
- What is the state of the common body of knowledge with respect to this technology at the start of the project?
- What efforts were taken to find out about the current state of the technology?
- Does the firm have access to the data and know-how ?
- What restrictions to access of technical data were present in determining the state of the art?

Examining the Technological Advances Sought

- What process was required to improve the state of the technology?
- How will success be defined & measured?
- How will technical conclusions be identified?
- What was learned from this project (or want to learn from this project)?
- What technical knowledge was learned from this project that will be protected from competitors?
- What IP came out or is expected to arise from this project?
- What would be done differently if it was to be done again?
- What is the team proud of in terms of technical achievement in this project?

Advancement

- What technological reason if any has prevented this firm or others from attempting the same thing previous to this development effort?
- What is out there and how does/will this resulting technology compare/differ to other

companies?

- Did the technological objectives change over the course of the project if so, how & why ? What were they unable to get working?

Technological Uncertainty

- Identify the limitations/constraints imposed by the technology components being utilised. What technical challenges did these constraints create?
- What degree of control does the claimant have to modify the technology components. What technical challenges did these constraints create?

TU – Determinism in the Technological Environment

- Is the integrated performance of multiple software components incorporated within the project fully deterministic?
- Is the vendor(s) capable of providing a deterministic description of the components predicted response when used in a unique fashion?

TU – Examining Alternatives

- What technical alternatives were examined, what was discarded & why?
- What are the technical design trade-offs associated with these alternatives?
- What are the possible technical outcomes other than the results being sought?

TU – A retrospective look

- What technology risks/constraints/problems appeared after the project began?
- What was hard or technically difficult to do & why ?
- Is the problem commonly recognised or being encountered as a critical challenge by other companies trying to do something that is similar?
- If it had to be done again what would be done differently?

Sufficiency

Sufficiency

- **Natural work product** usually produces sufficient evidence pertaining to eligibility.
- The merits of a claim should be examined from the highest level or the most general level of abstraction possible.
- Work product can provide sufficient direct or indirect information for an independent, knowledgeable **practitioner of the art** (**PAs**) to form a reasoned opinion.

Sufficiency:
a reasonable person test

- Practitioners of the Art (PAs) are

- experienced software developers who actively work with the technologies in question and understand the SR&ED program.

- It is the consensus among PAs regarding what is eligible and sufficient - not an individual's opinion - that is most important.
  - Experienced software specialists (SAs) can successfully identify eligibility on the basis of evidence of the historic technology base and how it is to be or is advanced.
    - SAs must have a thorough understanding of the program,
    - A group of PAs provide a more reliable judgement,
    - Neutrality and objectivity are very important.

Sufficiency: a shared responsibility

- Ultimately, the claimant is responsible for
  - supporting the company's assertions,
  - having sufficient direct or indirect evidence to support the claim and explaining it, and
  - explaining the original status of the underlying technologies and the central technological problems or challenges.

- The CCRA's staff is responsible for:
  - understanding how the full range of natural, direct and indirect evidence can be effectively used to support the claims, and
  - explaining it to the claimants.

Sufficiency: Questions arising from the T661

1. Which documents support the assertions about the technological uncertainty?
2. Which documents support the assertions about the technological advancement?
3. Which documents support the assertions about content criterion?
4. What is the highest level or most general type of information that provides effective evidence?

Sufficiency: Documentation

Minimum Documentation Contains:

- evidence of identifiable central constraints, technological problems or challenges associated with an uncertainty,
- evidence that technological advancements were being pursued.
- Evidence of what the historical base of technologies could do,
- Evidence of what was attempted or achieved in the sense of the technology not the product.

Sufficiency: Justifying the size of a claim

Reviewers should be able to see that the efforts on the central constraints, challenges,

problems or uncertainties

- were commensurate with and
- directly related to the need to resolve the constraints, and
- appropriately related to the size of the claim.

The Project Description
What Is its Role?
The Project Description
3 - 4 Pages

- Role:
  - the claimant's expert opinion as to why the project qualifies for ITCs within the SR&ED legislation
    - identifies S/T Advancement sought and the associated S/T Uncertainties,
    - identifies the major activities undertaken to resolve these Uncertainties and seek the Advance,
    - ties to contemporaneous supporting documentation.

The Elements

- Specifically required by the T661:
  - A. Scientific or Technological Objectives
  - B. S/T **Advancement**
  - C. S/T **Uncertainties**
    - to be resolved to achieve **Advancement**
  - D. **Work** within tax year
    - to resolve the **Uncertainties**
  - E. Documents available to substantiate **Work**
- The Fiscal Year
  - declare the fiscal year,
  - it helps to include a footer on each page beyond the first showing the project title and fiscal year.
- Project name
  - try to name the project for the technical issue rather than for a product,
  - however, product name can be used as the project name.
- Project Number
  - accounting reference(s).
- Start Date
  - date at which first eligible work was done.
- Actual or Estimated Completion Date
  - estimated completion date if the project is incomplete at the end of the fiscal year. Otherwise, date of last eligible work.
- Background
  - project background sufficient to allow a science advisor to understand the business

environment in which this software was developed and the business environment for which this software was to be used,
  - Explanation of the base technologies, their technical capabilities and limitations,
  - **This material may be included as part of the Technical Objectives section rather than shown separately.**
- S/T Objective
  - define success (quantitatively or verifiably).
- **S/T Advance**
  - New knowledge sought - i.e. the knowledge which you would be willing to present with pride to a peer. Your proof that this is an advance e.g. reference 3rd party product, your prior product technical capabilities or trade journal article reference to available capabilities.
- **Uncertainties [linked to the S/T Advance]**
  - list a few key central uncertainties, challenges, constraints associated with the Advance sought,
  - the technical issues which worried you most at the outset and possible alternative solutions,
  - cite your initial approach to the solution of each uncertainty (may not be the final approach).
- Activities and Progress **(Work)**
- **Briefly state Methodology**
  - in the language of the project: design, develop, test, redesign, redevelop, retest as is relevant to the project,
  - if you have an established methodology, make reference to it.
- Activities and Progress **(Work)** continued
  - **list in chronological order with at least approximate dates the major events, activities, tests and test results relevant to the Advance and Uncertainties,**
  - **new directions as a result of tests (iteration),**
  - **Status at year end**
    - what uncertainties were resolved during the year.
    - Outstanding uncertainties. The extent, if any, to which the Advance has been made.
- Documentation
  - a short list (3-10) of contemporaneous documents that support the key statements made in the technical description,
  - title, date and prime author of each document.
- Personnel
  - name, qualifications and job of project prime i.e. the first technical individual who might be interviewed by CCRA during any technical review.

<u>Scope of a SR&ED Project</u>
Beginning, size, end and the slide
<u>The SR&ED Project</u>

- Often not the same scope as the commercial project
- Can be a subset of the commercial project when multiple components are being developed and integrated that are based on established software design techniques.
- Identify the core SR&ED rich efforts as the starting point for scoping the claim.

Beginning

- In principle Eligibility begins:
  - when technological uncertainty is first identified
- **However there is no eligible activity if the project is only planned and never actually seeks an Advance**

The End

- Project only ends when
  - the attempt is abandoned,
  - the Advance is consistently demonstrable within the context of the constraints associated with the technological objectives.

The Slide

- With commercialization
  - development team gradually undertakes support work for customers and marketing,
  - development team starts working on future releases within the same architecture using the same tools without further Advance or Technological Uncertainties.
- We would expect
  - to see some transition of development personnel to ineligible work at or somewhat before first productive use
  - a clear statement of unresolved uncertainties is needed if the SR&ED project continues past first productive use
  - ongoing charges commensurate and directly related to the outstanding uncertainties are still in the domain of the original SR&ED project

The Size

- All analysis, design, engineering, data collection, programming, experimentation and testing needed in developing and proving-out the SR&ED are included in the SR&ED project.
- Components, features, integration, integration testing etc. needed to prove-out the central solutions, constructs, architectures, etc. associated with the SR&ED are included.

Sizing

- Identify central SR&ED efforts at the highest level which capture the effort and identify associated component developments where appropriate.
- Identify the key associated developmental activates for the central SR&ED efforts at the highest appropriate level.

Note: appropriate costing methodologies must be in place to isolate work by development personnel on other non-SR&ED activities or components.

Sources & Usefulness of Evidence

Software Work Product & Evidence Supporting Eligibility

- Numerous forms of sufficient indirect documentation usually arise naturally from a given Software development process.

Forms of Evidence are affected by:

- Business environment,
- Corporate maturity,
- Specific development methodologies used,
- Specific technologies being used,
- Pace of marketplace.

Characteristics of Modern Software Development Methodology

- Exists in accelerated timelines,
- Design iterations occur on the scale of hours & days NOT weeks & months,
- Requires highly parallel life cycle phases:
  - development,
  - deployment,
  - support & maintenance,
- Increased dependency on multiple 3rd party platforms,
- Resulting technology has a very short shelf life.

Basic Project Environments

- Distinct, well defined individual SR&ED projects
  - Clear start & finish
  - Narrowly defined technological objectives and uncertainties
  - By definition easily maps to administrative components of SR&ED

- Dedicated environments
  - Large dedicated R&D departments
  - Large focus on research may allow non SR&ED to be exception

- Large, mixed and/or multi-year projects
  - Usually contains mixture of eligible SR&ED as well as routine development
  - Requires extraction of eligible activities
- Early stage/start-up
  - Pre-revenue- may present similarly to dedicated research group
  - Post revenue - usually contains mixture of eligible & ineligible within a given project.

Corporate Development Indicators

- **Quality systems**
  - ISO
  - CAR systems
  - Defect tracking systems
  - Configuration management

  - Formal release processes

- **Published SE methodology standards**
  - Design reviews
  - Coding standards
  - Best practice architecture standards

Key Evidence recognition factors

- Accommodating the fundamental development environments dictated by the marketplace,

- Regular extraction of SR&ED eligible work from a larger development context,

- By convention and practice, the preferred embodiment of evidence is in a high level, summarised, and indirect form.

Challenges

- Ensuring size of claim is commensurate with the project scope.
- Interpretation and sufficiency of evidence.
- Efficiently estimating/identifying and allocating eligible expenditures.

Role of Software T661 Project Descriptions

- Describe/identify the eligible components/aspects claimed from within a given development effort,
- Not meant to be exhaustive or tutorial in nature ,
- Functions as the “Technological lens”
  - Directs reviewers to “focus on and interpret” selective components of naturally arising documentation,
- Forms a high level summary of taxpayer’s expert opinion of eligibility.

Work Product (Evidence)

- Developer’s journals,
- Architectural design docs,
- Design review minutes,
- Written correspondence (letters, email, fax),
- Performance requirements.
- Simulations
- Defect tracking
- Test results
- 3rd party documentation
- Detail design documentation

- Test plans
- Project schedules
- Product requirements
- Source code & revision control
- Build reports

Evidence Sources Supporting Financial Claims

- Project Schedules
  - Key resources, key activities
  - Materials & expense tracking, calendar duration,
  - Caution many show planned not actual
  - Often constructed in context of a larger dev effort
  - Useful to identify & extract eligible components from larger context

- Resource Allocation Plans
  - Prevalent in Dedicated research environment

- Time Sheets
  - Caution on level of granularity
  - Electronic entry becoming very common, paper - almost gone
  - Effectiveness depends on specific accounting systems, larger ones less flexible
  - Granularity of reporting is usually at a high level
    - Overhead of tracking changing & emerging activities on a detailed individual basis usually not feasible

- Technical Supervisor's Summaries

  - Done weekly, monthly or event based
  - Very brief, a few sentences per entry
  - Provides scope of effort/activities at highest level
  - Identifies resources/expenses deployed
  - Industry reluctant to adopt
    - Tendency for CCRA to ask for increasingly finer level of detail each year.

In Summary

- There exists a wide selection of naturally arising work product & document sources to provide sufficient evidence to support most eligible claims.
- It is crucial that the evidence be examined from the highest level.
- The T661 plays a critical role in framing the context in which the evidence will be examined.
  - Serves as the taxpayer's expert opinion
  - Serves as the technological lens with which to view evidence

Best Practices

Best Practice Objectives

- Identify OPTIONAL techniques that:
  - Improve the clarity of the naturally arising evidence,
  - Use simple, low overhead practices,
  - Are available to all software development methodologies.

Suggestion: Birth certificate

- Done at commencement of a project,
- Very terse, condensed project outline identifying:
  - Major constraints and technological challenges,
  - Potential technological uncertainties,
  - Potential technological advancements sought,
  - Potential approaches –proven and unproven.
- Very short, skeletal, dated,
- Can become the starting point for a contemporaneous T661.

Suggestion: SR&ED Flagging/Tagging

- Set-up formal Filter/Tagging system to identify eligible projects or subprojects,

- Set-up formal process to have the development team contemporaneously identify natural work product that is useful as SR&ED evidence.

Suggestion: SR&ED Flagging/Tagging

- Place "tags" into current processes and create archive,
  - E.g. on Defect tracking records
    - incorporate an attribute into DT databases enabling the issues entered to be "tagged" as SR&ED related,
  - E.g. E-mail
    - create a common e-mail account in which any correspondence with SR&ED significance can be "CC'd". The result is a single repository of all SR&ED related e-mail for a given project.

Suggestion: Status Report Augmentation

- Add notation with respect to major events related to TUs and TAs for ITC eligible project,

- Collect and archive on regular basis
  - monthly, quarterly, or at development gates.

Suggestion: Status Report Augmentation

- Can take various forms
  - design reviews adding focus covering major events, TU & TA elements,

- Regular status reports augmented to meet both business requirements and the requirements for SR&ED eligibility,
- Separate, contemporaneously updated draft of the T661.

Prototype Archiving

- Archive a full copy of each key prototype complete with:
  - execution environment,
  - test data.

- Place emphasis on saving key revisions which address the major issues associated with the SR&ED components of a given development project.

Some Suggestions: Expenditure Tracking

- **Resource Allocation Plans**
  - identify resources associated with key specific activities,
  - where the activities can be delineated on the basis of ITC eligibility,
    - use these planning records to frame an estimate of the total time spent on R&D activities.

Some Suggestions: Expenditure Tracking

- Time Sheets
  - Individual time sheets which separate time spent on SR&ED work from non-SR&Ed work,
    - Aggregate as highly as possible, commensurate with obtaining separation of SR&ED work from material non SR&ED work.

Suggestions: Expenditure Tracking

  - **Supervisors Summaries**
    - **P**eriodic weekly, monthly, quarterly,
    - Constructed by technical supervisors in “real time”,
    - Estimate the amount of SR&ED work undertaken for the given period as well as any major ineligible activities during the period in question,
    - Can use negative reporting system where proper filters exist.

In Summary

- Establish the base from which the work began, and against which advance is shown
- Look for and preserve the evidence
- Real communication is key - labels are unreliable

Contact information:
Kay James
CCRA - National Technology Sector Specialist, Software/IT
613-941-1109
kay.james@ccra-adrc.gc.ca

CATA*Alliance*
613-236 -6550