

Doc No: SC22/WG21/N1783  
J16/05-0043  
Date: Apr 9, 2005  
Project: JTC1.22.32  
Reply to: Herb Sutter  
Microsoft Corp.  
1 Microsoft Way  
Redmond WA USA 98052  
Email: hsutter@microsoft.com

## **TG5 Liaison Report #9**

Meeting #10 of Ecma TC39/TG5 (C++/CLI) was held in Kamuela, HI, USA, on March 7–8, 2005.

---

## **Agenda**

**for the:**

**10<sup>th</sup> meeting of Ecma TC39-TG5**

**to be held in:**

**Kamuela, HI, USA**

**on:**

**7,8,11 March 2005**

### **TIME:**

09:00 till 17:00 on Mon 7<sup>th</sup> March 2005  
09:00 till 17:00 on Tue 8<sup>th</sup> March 2005  
09:00 till 10:00 on Fri 11<sup>th</sup> March 2005

[8:30AM Breakfast, Noon lunch each day]

### **LOCATION:**

Fairmont Orchid Hotel and Resort  
1 N. Kaniku Drive  
Kamuela, Hawaii, USA  
96743  
+1 808-885-2000

(Directions and Maps below)

### **CONTACT:**

Thomas Plum  
[tplum@plumhall.com](mailto:tplum@plumhall.com)

## **1 Opening**

- 1.1 Appointment of Recording Secretary**
- 1.2 Introduction of participants**
- 1.3 Host facilities/local information**

## **2 Adoption of the agenda (TG5/2005/008)**

## **3 Final approval of minutes of previous TG5 meeting (TG5/2005/007)**

## **4 Matters arising from the minutes not covered elsewhere**

## **5 Project Editor's Report**

## **6 Approving tracked changes in latest draft**

## **7 Date and place of next meetings**

- 7.1 Mon Sep 19, Tue Sep 20, Fri Sep 23 (9-10am), Redmond WA**

*NOTE*

*TC39 business meeting takes place Sep 23 (PM)*

## 8 Reports from Liaisons

8.1 TC39 TG3 (CLI) – Rex Jaeschke

8.2 SC22/WG21 (C++) – Tom Plum, P. J. Plauger, Tana Plauger, John Spicer, and Steve Adamczyk

8.3 TC39 TG2 (C#) – Rex Jaeschke

## 9 Action item spreadsheet review

## 10 Approval of TG5 spec to forward to TC39

## 11 Any other business, and appreciation of hosts

## 12 Adjournment

### DIRECTIONS from KOA:

**Starting from:** **A** Keahole Kona International Airport KOA, Kailua/kona, HI

**Arriving at:** **B** 1 N Kaniku Dr, Kamuela, HI 96743-9731

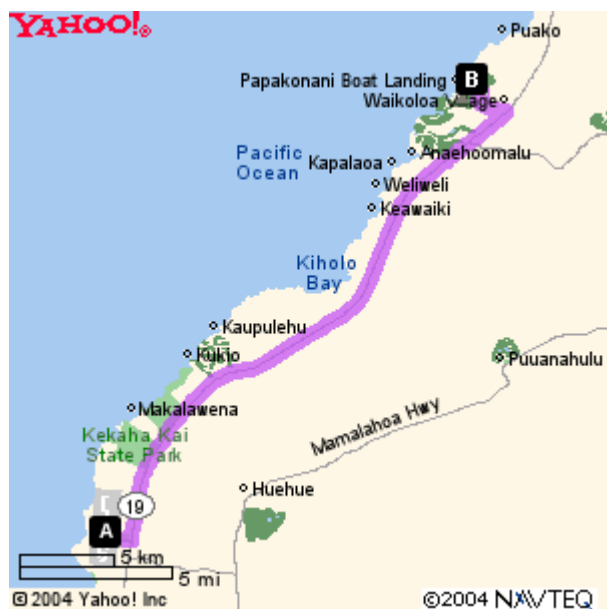
**Distance:** 22.3 miles    **Approximate Travel Time:** 42 mins

### Your Directions

1. Start going towards the **AIRPORT EXIT** on **KEAHOLE KONA** - go **0.2** mi
2. Continue on **KOA** - go **< 0.1** mi
3. Continue on **KEAHOLE KONA** - go **0.6** mi
4. Bear **L** on **HI-19** - go **19.5** mi
5. Turn **L** on **MAUNA LANI DR** - go **1.0** mi
6. Turn **R** on **KANIKU DR** - go **0.9** mi
7. Arrive at **1 N KANIKU DR, KAMUELA**, on the **R**

When using any driving directions or map, it's a good idea to do a reality check and make sure the road still exists, watch out for construction, and follow all traffic safety precautions. This is only to be used as an aid in planning.

### Your Full Route



### Your Destination



**Address:**  
1 N Kaniku Dr  
Kamuela, HI 96743-9731

This is a replacement/place-holder for Document TC39-TG5/2005/009, “Working Draft 1.10 of the C++/CLI Standard, Language”. This draft can be found at the following URL: <http://www.plumhall.com/ecma/index.html>.

	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
25	16-Dec-03	Phone meeting	10	Technical	H	Brandon Bray	Revise this clause by covering topics including application entry point, assembly boundaries, among others.		No	
36	16-Dec-03	Phone meeting	15.3.8	Technical	M	Brandon Bray	cv-qualification needs to be considered for <code>dynamic_cast</code> .		No	
43	16-Dec-03	Phone meeting	15.11.1	Technical		Mark Hall	Add support for handle equality comparison, and handle <code>==/!= nullptr</code> , and vice versa.	<p>Meeting #3 (Mel): Had a short discussion. Mark will produce a paper for the May meeting.</p> <p>Meeting #4 (NJ): No progress. To be discussed via email, and at the Jun meeting</p> <p>Meeting #5 (WA): Discussed briefly. Asked Mark to write this up and distribute to the reflector.</p> <p>Phone call Jun 29: This issue was resolved; just needs drafting of final words.</p> <p>Meeting 7 (WA): In the case of <code>if(handle)</code>, which conversions are attempted before comparison against <code>nullptr</code> is used?</p> <p>We agreed that if an explicit conversion to <code>bool</code> exists, <code>if(handle)</code> uses that.</p> <p>There is no implicit unboxing.</p> <p>Steve and Mark worked on this and presented it to the full committee on the 2nd day.</p> <p>Based on committee feedback, Mark will write this up for future consideration.</p>	No	
47	16-Dec-03	Phone meeting	17	Technical	M	Brandon Bray	Provide text for this clause (Namespaces)		No	
50	16-Dec-03	Phone meeting	18.4	Technical	M	Brandon Bray	Extend declarator-id's by adding a new production that allows default.		No	
58	16-Dec-03	Phone meeting	18.6.6.1	Technical		Mark Hall	Reword this subclause similarly to the way special member functions are described.	Meeting 7 (WA): ?? To be done in Tue morning work sessions.	No	
62	16-Dec-03	Phone meeting	18.10.1	Technical	L	Brandon Bray	Add a description that for any value class we have to make the copy before calling member functions.	Meeting #9 (NJ): Needs to be done.	No	
66	16-Dec-03	Phone meeting	21	Editorial	M	Brandon Bray	Introduce value classes -- Discuss the following: value classes are optimized for small data structures. As such, value classes do not allow inheritance from anything but interface classes. Tie in fundamental classes.		No	
67	16-Dec-03	Phone meeting	21.4.1	Technical	H	Brandon Bray	Add words about instance constructors and static constructor. Value classes cannot have SMFs (specifically, default constructor, copy constructor, assignment operator, destructor, or finalizer. Need to add specification for this along with rationale.		No	
74	16-Dec-03	Phone meeting	23.5	Technical	M	Brandon Bray	Write-up array covariance w.r.t arrays.		No	
75	16-Dec-03	Phone meeting	23.6	Technical	M	Brandon Bray	Write up array initialization.		No	
76	16-Dec-03	Phone meeting	24.4	Technical	H	Brandon Bray	Address what happens when a ref class does not implement an interface function (and what happens when a base class has a non-virtual function with the same name).		No	

	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
79	16-Dec-03	Phone meeting	27	Technical	H	Brandon Bray	Cover unification of CLI and Standard C++ exception handling models, and anything else that might go in this clause.  Are exceptions asynchronous now in some cases? Yes they are. (For example, <code>NullReferenceException</code> .)	Meeting #5 (WA): Kevin Free (Microsoft) gave a verbal presentation.  <code>catch(...)</code> catches managed and native exceptions.  <code>catch(System::Object^)</code> also catches both kinds, but won't invoke the destructor (so can leak).  CLI exception handling supports more features than we expose.  The issue remained with Brandon to write up, as before.	No	
82	16-Dec-03	Phone meeting	29	Technical	M	Brandon Bray	Flesh out "Templates" clause.		No	
94	29-Jan-04	meeting #2 (HI)		Technical		Mark Hall	Relationship between primitive types and CLI types.  The current spec allows the following: <code>int i = 10; String^ s = i.ToString();</code> Standard C++ doesn't allow member selection on expressions of primitive type. Assuming <code>int</code> maps to <code>System::Int32</code> , just how much alike are these two types? Specifically, when do we treat the primitive as the underlying class.	Meeting 5 (WA): Asked Mark to write this up and distribute to the reflector. Please address the side-effect issue; that is, given <code>(i++).ToString()</code> , is the increment done?  Meeting 7 (WA): ?? To be done in Tue morning work sessions.  Re the side-effect, yes, it must be done.	No	
98	29-Jan-04	meeting #2 (HI)	30	Technical	R	Brandon Bray	Restrictions on generics re generic code generation.  The current generics clause needs to be fleshed out, especially w.r.t how overload resolution works within the CLI.	Meeting #2 (HI): Brandon will write a paper on this.  Meeting #4 (NJ): The fleshing out of Clause 30 is a significant contribution toward this. More work needed in declarations and function calls.	No	
105	29-Jan-04	meeting #2 (HI)	14.5.1	Technical		Mark Hall	Constructors can't be used in casts in managed classes. Should they be allowed in explicit conversions? All managed type constructors being explicit by default. (Already yes, but reconfirm this.)	Meeting #4 (NJ): Steve will send the editor sufficient text to go into the public drop to indicate our intention re this topic. DONE.  Meeting 5 (WA): Asked Mark to write this up and distribute to the reflector.  Meeting 7 (WA): Steve and Mark worked on this and presented it to the full committee on the 2nd day. Mark will write this up for future consideration.	No	
111	19-Feb-04		15.3.2	Technical	M	Brandon Bray	Need to consider how indexed access expressions are interpreted in templates.		No	
117	19-Feb-04		18.4.2	Technical	H	Brandon Bray	The qualified name of a property needs to be described somewhere. Once that happens, how an out-of-class definition is done will already be covered by existing rules.		No	

	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
124	10-Jun-04	Jonathan Caves		Technical		Jonathan Caves	<p>Indexed properties -- Consider the following:</p> <pre> interface class I1 {   property int Value; };  interface class I2 {   property int Value[String^] {     int get(String^);     void set(String^, int);   }; };  ref class D : I1, I2 {   // Implements the properties };  D^ d; d-&gt;Value["Foo"];  The question is what does the last line do?  Which leads to a language design question - what should the compiler do when faced with a property followed by a '['  1) Should it look for just parameterized properties and if there isn't one fail - I suspect not  2) Should it look for all properties and if the returned set contains a parameterized property it should prefer it - this sounds like magic to me.  3) Should it look for all properties perform overload resolution across the whole set and if the resulting call is ambiguous then issue an error. </pre>	<p>Meeting #5 (WA): Discussed this. Option #3 preferred.</p> <p>Meeting 7 (WA): Discussed this in detail.</p> <pre> property int Value[int] {   void set(int, int); };  x-&gt;Value[1] = 4 is treated as x-&gt;set_Value(1,4);  -----  property array&lt;int&gt;^ Value {   array&lt;int&gt;^ get(); }  x-&gt;Value[1] = 4 is treated as x-&gt;get_Value()[1] = 4  -----  property int% Value[int] {   int% get(int); }  x-&gt;Value[1] = 4 is treated as x-&gt;get_Value(1) = 4  This construct violates the principle of properties (that of setting/getting the value of some property), so is not to be encouraged; however, it is supported, but no need to consider it further here. </pre>	No	
125	14-Jun-04	meeting #5 (WA)	8.15.3	Technical	M	Brandon Bray	<p>Based on the rules for type deduction in templates, it seems surprising that you can match <code>array&lt;ItemType&gt;^</code> with an argument of type <code>int</code>. Here is a standard C++ example intended to illustrate the issue:</p> <pre> template &lt;class ItemType&gt; struct Stack {}; template &lt;class ItemType&gt; struct Array {   Array(ItemType); };  template &lt;class ItemType&gt; void PushMultiple(Stack&lt;ItemType&gt;, Array&lt;ItemType&gt;);  int main() {   Stack&lt;int&gt; s;   PushMultiple(s, 1); // deduction fails   PushMultiple&lt;int&gt;(s, 1); } </pre> <p>Are the rules for generic different in this area?  [There seems to be information related to this in 30.3.2. See that subclause for further comments on this issue.]</p>		No	



	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
132	14-Jun-04	meeting #5 (WA)	15.3.10	Technical	M	Brandon Bray	Unboxing and boxing are described as preferred user-defined conversions; however, this is incorrect.		No	
138	14-Jun-04	meeting #5 (WA)	18.4	Technical		Mark Hall			No	
143	14-Jun-04	meeting #5 (WA)	19.7	Technical	L	Brandon Bray	The restriction below does not apply to non-static member operators – that need not have a parameter of the type of the class.	Meeting #9 (NJ): Needs to be done.	No	
151	14-Jun-04	meeting #5 (WA)	25.2	Technical	M	Brandon Bray	The note says "pickup the restrictions from page 333 (of Brandon's paperback copy of the C# spec)".		No	
153	14-Jun-04	meeting #5 (WA)	30.1	Technical	M	Brandon Bray	The text indicates that a generic-declaration may appear in a class scope, but the syntax of member declaration has not been extended to permit a generic-declaration. [[#98]]		No	
154	14-Jun-04	meeting #5 (WA)	30.1	Technical	R	Brandon Bray	Doesn't the text "a generic name declared in namespace scope or in class scope shall be unique in that scope" make the first sentence of this paragraph redundant? Re the reference to 14.5.4: That is the section on partial specialization. Generics can't be partially specialized, can they? The spec. should probably answer that explicitly.		No	
155	14-Jun-04	meeting #5 (WA)	30.1	Technical	R	Brandon Bray	What is a non-generic type? Does it mean that the rules are the same as classes? As template classes? Something else?		No	
158	14-Jun-04	meeting #5 (WA)	30.1.1	Technical	R	Brandon Bray	The equivalent wording for template parameters in the working paper has been changed to "defines its identifier to be a typedef-name". The revised wording should probably be used here too (see core issue 283)		No	
160	14-Jun-04	meeting #5 (WA)	30.1.6	Technical	R	Brandon Bray	This subclause describes when a static constructor is invoked. In 18.8, it references the CLI Standard Partition II (10.5.3). Are the rules the same? (Yes) Should this subclause also just reference the CLI spec? There are two sets of behavior; we need to say which one we use.		No	
161	14-Jun-04	meeting #5 (WA)	30.1.7	Technical	M	Brandon Bray	What to say about explicit conversion functions (which can only occur in managed class types)?		No	
162	14-Jun-04	meeting #5 (WA)	30.2.2	Technical	R	Brandon Bray			No	
							This subclause lists the types that can and cannot be generic arguments. Fundamental types are not included in either set, neither are function types. The subclause does not say whether or not cv-qualified types are allowed.			

	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
167	14-Jun-04	meeting #5 (WA)	30.3	Technical	L	Brandon Bray	<p>"When the type of a parameter or variable is a type parameter, the declaration of that parameter or variable shall use that type parameter's name without any pointer, reference, or handle declarators."</p> <p>What about cv-qualifiers?</p>	Meeting #9 (NJ): Needs to be done. CV-qualifiers are not permitted.	No	
169	14-Jun-04	meeting #5 (WA)	30.3.2	Technical	L	Brandon Bray	<p>The issue raised in 8.15.3 is somewhat answered here. 18.3.6 seems to deal with expanded forms of calls, not expanded forms of function declarations. I interpret the text above as saying that deduction is done as if the function were declared like this:</p> <pre>generic &lt;typename ItemType&gt; void PushMultiple(Stack&lt;ItemType&gt;^, ItemType i1, ItemType i2,/* ... */);</pre> <p>Is that correct? I think this requires a more detailed description.</p>	Meeting #9 (NJ): Needs to be done. Add example(s).	No	
170	14-Jun-04	meeting #5 (WA)	30.3.2	Technical	L	Brandon Bray	<p>Something needs to be said about instantiating a generic delegate using a generic function.</p>	Meeting #9 (NJ): Needs to be done.	No	
171	14-Jun-04	meeting #5 (WA)	30.4.2	Technical	H	Brandon Bray	<p>When are members considered hidden? Is it using the rules described later? Those are described as applying only when a type parameter has both a class constraint and one or more interface constraints though.</p>	Meeting #9 (NJ): Needs to be done.	No	

	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
172	14-Jun-04	meeting #5 (WA)	30.4.4	Technical	H	Brandon Bray	<p>Miscellaneous generics issues:</p> <p>1. I seem to recall discussions of other kinds of constraints (I believe one of them concerned whether you could do a "new T()").</p> <p>2. Doesn't there need to be some discussion of how overload resolution works when a function argument has a type parameter as its type?</p> <p>3. Are the typename and template rules for syntactic disambiguation the same in generics as in templates? Presumably, the lack of specialization would eliminate the need for these.</p> <p>4. If scope contains a set of overloaded generic functions, is partial ordering used to choose between them?</p> <p>5. I assume since there is nothing that says otherwise, that generics can be friends of other classes and generics can make other classes, functions, (including generics) friends?</p> <p>6. If friendship is supported, can a generic first be declared in a friend declaration (suggested answer: no).</p> <p>7. Standard C++ has restrictions on type parameters such as prohibiting types with no linkage. Does this rule apply to generic arguments?</p> <p>8. Are there generic conversion functions?</p>	<p>Meeting #8 (WA):</p> <p>1. For V1, we can consume and enforce these special constraints, but we can't author them. However, we plan to do so in future, so add this to "Future directions".</p>	No	
173	14-Jun-04	meeting #5 (WA)	32.1.4	Technical	L	Brandon Bray	<p>To ensure that signatures for the same Type produced by different implementations match, the ordering in such a set of modreqs and modopts is as follows: first modreqs in ascending order by name, then modopts in ascending order by name, with case being significant. [[We need some rule here; is this the one?]].</p>	<p>Meeting #9 (NJ): Add a description of our best guess at the correct solution, to Future Directions, then mark this Postponed. Point to this from the normative text somehow.</p>	No	

	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
184	2-Aug-04	meeting #6 (WA)		Technical		Herb Sutter	Describe problem with overloading on % vs. &  Herb presented the following code:  <pre>#include &lt;iostream&gt; using namespace std; void f( const int&amp; ) { cout &lt;&lt; "f( const int&amp; )" &lt;&lt; endl; } void f( int&amp; )      { cout &lt;&lt; "f( int&amp; )" &lt;&lt; endl; }  void g( int% )      { cout &lt;&lt; "g( int% )" &lt;&lt; endl; } void g( int&amp; )      { cout &lt;&lt; "g( int&amp; )" &lt;&lt; endl; }  int main() {     const int ci = 0;     int i = 0;     int^ hi = gcnew int;      f( ci );     f( i );      g( *hi );     // g( i );    // ambiguous: should g(int&amp;) be preferred? }</pre> <p>The following code was his attempt to write an agnostic swap:</p> <pre>template&lt;typename T&gt; void swap( T% a, T% b ) {     #if defined NO_PIN_PTR          // doesn't work         T temp = a; a = b; b = temp;     #elif defined PIN_PTR_BUG      // doesn't         compile         T temp = *pin_ptr&lt;T&gt;(a);         *pin_ptr&lt;T&gt;(&amp;a) = *pin_ptr&lt;T&gt;(&amp;b);</pre>		No	
194	2-Aug-04	Anthony Williams	15.3.2	Technical		Jonathan Caves	Re Anthony's post to the reflector re "default index	Meeting 7 (WA): Discussed the possibility of disallowing both the default indexed property and operator[].	No	
196	30-Sep-04	meeting #7 (WA)		Technical		Herb Sutter	In native types, % behaves like &.		No	
198	30-Sep-04	meeting #7 (WA)	2	Technical		Herb Sutter	Propose wording to require that extensions over and above ISO C++ requirements, be diagnosed.	Meeting 9 (NJ): Re the new paragraph added to §2. "Conformance" in response to spreadsheet issue #198, the committee believed this text does not adequately address the issue. The editor was asked to remove it.  Ownership was transferred from Tom to Herb.	No	
203	26-Oct-04	Rex Jaeschke	10.1.2	Technical	M	Brandon Bray	[Note: The compiler needs to add typedef members to the class so that template code can use the return type or the parameter types. [[Need more explanation.]] end note]		No	
204	26-Oct-04	Rex Jaeschke	12.2.2	Technical	M	Brandon Bray	Write intro text.		No	

	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
205	26-Oct-04	Rex Jaeschke	15.5	Technical	H	Brandon Bray	<p>15.5 Explicit type conversion (cast notation)            The rules in the C++ Standard (§5.4/5) have been extended for C++/CLI by including safe casts before static casts.</p> <ul style="list-style-type: none"> <li>• a const_cast</li> <li>• a safe_cast</li> <li>• a safe_cast followed by a const_cast</li> <li>• a static_cast</li> <li>• a static_cast followed by a const_cast</li> <li>• a reinterpret_cast</li> <li>• a reinterpret_cast followed by a const_cast</li> </ul> <p>[Note: Standard C++ programs remain unchanged by this, as safe casts are ill-formed when either the expression type or target type is a native class. end note]</p> <p>Provide background on the expected behavior and rationale. (Get this from the updated casting proposal.)</p>		No	
206	26-Oct-04	Rex Jaeschke	21.4	Technical	M	Brandon Bray	Simple value classes: Flesh this out.		No	
207	26-Oct-04	Rex Jaeschke	24.2.5	Technical	H	Brandon Bray	<b>Interface member access: Write up.</b>		No	
210	4-Dec-04	Rex Jaeschke	29.5.1	Technical	M	Brandon Bray	There is confusion about DefaultMember attribute and IndexerNameAttribute. In the current implementation, it appears that the first one is exhibiting the behavior of the second one, and the second one is being emitted into metadata directly when it should be consumed by the compiler.		No	
213	4-Dec-04	Rex Jaeschke		Technical	M	Brandon Bray	13.3.3.2/4 of the C++ Standard has rules for pointer conversions, that need to be adapted to handles. Review this subclause and determine the changes needed for the C++/CLI spec.		No	
215	Feb-3-2005	Jeff Peil	13.1.1, 13.1.3	Technical	2	Brandon Bray	§13.1.1 and 13.1.3 disagree, one describes gc-lvalue->lvalue as a conversion for native types, the other describes it as never having gc-lvalues for these (they are always l-values) They need to be made <u>consistent</u> .		No	
218	Feb-10-2005	Rex Jaeschke	15.3	Technical	M	Brandon Bray	<p>Are the productions</p> <pre> postfix-expression . pseudo-finalizer-name postfix-expression -&gt; pseudo-finalizer-name </pre> <p>necessary, and, if so, should the "pseudo-" prefix be dropped?</p>		No	

## 2005-03 Project Editor's Report

### Rex Jaeschke

Ecma TC39-TG5 project editor

[rex@RexJaeschke.com](mailto:rex@RexJaeschke.com)

+1 703 860-0091

Working Draft 1.10 has been produced and distributed. The following work went into producing it:

1. I applied corrections resulting from the Redmond Jan meeting.
2. I applied corrections resulting from the phone meetings.
3. I incorporated Brandon's paper on "Destructors and Finalizers".
4. I closed out numerous action items.

#### Known Issues

1. There are 51 open issues in the spreadsheet.
2. There are a number of TODO items in the draft marked [[Ed.]]. These will be resolved after consultation with MS developers and once a new version of the compiler is supplied.

#### Post Processing

Once the final working draft has been produced, I expect to do the following before submitting it to the Ecma CC and GA:

1. Adopt the tracked changes in the final draft as accepted/modified at the final meeting.
2. Adopt any other changes accepted at the final meeting meeting.
3. Compile all examples.
4. Proof the cross-reference index.
5. Check that all links lead to the correct place.
6. Apply any last minute editorial changes reported.
7. Circulate the final draft to TG5 members for review.

## Proposed Timeline for ISO/IEC JTC 1 Fast-Track

Rex Jaeschke, Microsoft

[rex@RexJaeschke.com](mailto:rex@RexJaeschke.com)

2005-02-27

Based on my experience with the first round of standardization of the C# and CLI specifications, here's my suggestion for a timeline for the TC39 standards voted out of TC39 in March 2005.

1. Mar, 2005, TC39 agrees to forward the final draft based on the TG's recommendation.
2. Jun, 2005, the Ecma office will notify JTC 1 that Ecma expects to submit the spec via the Fast Track process in Jul, 2005, and provides an advance copy of the draft spec for circulation as a courtesy.
3. Jun, 2005, at its semi-annual business meeting, the Ecma General Assembly (GA) adopts the submission as an Ecma standard, Version x, gives it a number, and makes it available for free from the Ecma public website.
4. Jul, 2005, the Ecma standard is submitted to JTC 1 for Fast Track processing. JTC 1 determines that Subcommittee 22 (SC22 — programming languages and environments) is the appropriate home for this, and assigns the task to SC22.
5. Mid-Jul, 2005, SC22 starts a 6-month letter ballot period.
6. While National Bodies (NBs) are reviewing the specs and, ultimately, submitting comments, so too can the originating TG via Ecma. That TG might want to meet in person or have one or more phone conferences to determine what these comments are and possibly what should be its own formal response to those comments.
7. Jan 1, 2006, a JTC 1 ballot resolution meeting date, location, chairman, and project editor are proposed by Ecma.
8. Mid-Jan, 2006, SC22's 6-month letter ballot period ends and all comments are due to JTC 1's ITTF. (All comments must be submitted electronically using a specific Word template.)
9. Feb 1, 2006, JTC 1's ITTF collates all the ballots and their associated comments, and makes them available to the ballot resolution committee (which is, essentially, the originating TG).
10. Feb 1, 2006, the SC22 Secretariat announces the date and location of the ballot resolution meeting.
11. Feb 1–mid-Mar, 2006, the TG works on producing formal responses to all public comments.
12. Late Mar, 2006, the ballot resolution meeting is held for x days. Any NB that has voted NO on the ballot must send a representative; otherwise, their NO vote will be ignored. (Assuming that all or a sufficient number of the NBs voting NO convert that to a YES based on decisions made at the ballot resolution meeting, the draft is unofficially an ISO/IEC standard.)
13. Apr, 2006, the project editor integrates all changes based on the ballot resolution meeting, and forwards the revised spec to ITTF for final proofing and processing.
14. May, 2006, the corresponding Ecma standard is revised to match that adopted by ISO/IEC.

15. Sep, 2006, TC39 votes to forward this revised draft to the Ecma GA for adoption.
16. Late Sep, 2006, the spec is announced as an ISO/IEC standard.
17. Sep, 2006, at the annual ISO/IEC JTC 1/SC 22 plenary, I (as Ecma-to-SC22 liaison) request that JTC 1 make available for free, the ISO/IEC version of the standard.
18. Nov, 2006, JTC 1 approves the free availability.
19. Dec, 2006, the Ecma GA adopts Version x+1 of the standard, which, except for some typographical and front matter differences, is identical to that from ISO/IEC.

As you can see, the Fast Track process takes about 15 months, Jul, 2005 to Sep, 2006.

Ordinarily, Ecma TC39 and the GA deal with new or revised standards at their semi-annual meetings. (While this could be done via letter ballots, it's not the normal case.) As such, this timeline synchronizes with the Mar, 2005, and Jun, 2005, meetings of TC39 and the GA, respectively. If the Mar, 2005, date is missed, the whole schedule will slip 6 months, to synchronize with the next TC39 and GA meetings.



## ***TG5 Convener's Report to TC39***

***11 March 2005***

### **Officers**

Convener: Dr. Thomas Plum, Plum Hall Inc

Editor: Mr. Rex Jaeschke, Microsoft Corporation

### **Meetings**

The following meetings and phone conferences have occurred since the September 2004 report:

22-23 October 2004	Face-to-Face, Redmond, WA, USA, hosted by Microsoft
20-21 January 2005	Face-to-Face, Westfield NJ, USA, hosted by EDG and Dinkumware
3 February 2005	Phone conference
10 February 2005	Phone conference
17 February 2005	Phone conference
20-21 7, 8, 11 March 2005	Face-to-Face, Kamuela, USA, hosted by Plum Hall

The next upcoming face-to-face meeting is:

19,20, 23 September 2005	Face-to-Face, Redmond, WA, USA, hosted by Microsoft
--------------------------	---

### **Attendees**

The meetings were attended by representatives from member companies Borland, Dinkumware Ltd, Edison Design Group, IBM, Microsoft, and Plum Hall. Invited guests included representatives from Jagger Software.

### **Progress**

Over the last 6 months, the TG5 has had 3 face-to-face meetings and 3 phone conferences.

### **Latest Status**

The current working draft of the specification is WD1.10 (as of the March meeting). The TG agreed to push the finalization of edition 1 from March 2005 to September 2005. The TG did not want to rush a specification that was not ready. As of 8 March 2005, there are 50 outstanding technical action items, and 1 editorial action item, some of which just need an official re-visit to resolve. The convener would like to thank the task group for all of their hard work during the last six months.

### **Drafts to be submitted to TC for adoption**

None

Thomas Plum

**Convener TG5**

	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
25	16-Dec-03	Phone meeting	10	Technical	H	Brandon Bray	Revise this clause by covering topics including application entry point, assembly boundaries, among others.		No	
36	16-Dec-03	Phone meeting	15.3.8	Technical	M	Brandon Bray	cv-qualification needs to be considered for <code>dynamic_cast</code> .		No	
43	16-Dec-03	Phone meeting	15.11.1	Technical		Mark Hall	Add support for handle equality comparison, and handle <code>==/!= nullptr</code> , and vice versa.	<p>Meeting #3 (Mel): Had a short discussion. Mark will produce a paper for the May meeting.</p> <p>Meeting #4 (NJ): No progress. To be discussed via email, and at the Jun meeting</p> <p>Meeting #5 (WA): Discussed briefly. Asked Mark to write this up and distribute to the reflector.</p> <p>Phone call Jun 29: This issue was resolved; just needs drafting of final words.</p> <p>Meeting 7 (WA): In the case of <code>if(handle)</code>, which conversions are attempted before comparison against <code>nullptr</code> is used?</p> <p>We agreed that if an explicit conversion to <code>bool</code> exists, <code>if(handle)</code> uses that.</p> <p>There is no implicit unboxing.</p> <p>Steve and Mark worked on this and presented it to the full committee on the 2nd day.</p> <p>Based on committee feedback, Mark will write this up for future consideration.</p>	No	
47	16-Dec-03	Phone meeting	17	Technical	M	Brandon Bray	Provide text for this clause (Namespaces)		No	
50	16-Dec-03	Phone meeting	18.4	Technical	M	Brandon Bray	Extend declarator-id's by adding a new production that allows default.		No	
58	16-Dec-03	Phone meeting	18.6.6.1	Technical		Mark Hall	Reword this subclause similarly to the way special member functions are described.	Meeting 7 (WA): ?? To be done in Tue morning work sessions.	No	
62	16-Dec-03	Phone meeting	18.10.1	Technical	L	Brandon Bray	Add a description that for any value class we have to make the copy before calling member functions.	Meeting #9 (NJ): Needs to be done.	No	
66	16-Dec-03	Phone meeting	21	Editorial	M	Brandon Bray	Introduce value classes -- Discuss the following: value classes are optimized for small data structures. As such, value classes do not allow inheritance from anything but interface classes. Tie in fundamental classes.		No	
67	16-Dec-03	Phone meeting	21.4.1	Technical	H	Brandon Bray	Add words about instance constructors and static constructor. Value classes cannot have SMFs (specifically, default constructor, copy constructor, assignment operator, destructor, or finalizer. Need to add specification for this along with rationale.		No	
74	16-Dec-03	Phone meeting	23.5	Technical	M	Brandon Bray	Write-up array covariance w.r.t arrays.		No	
75	16-Dec-03	Phone meeting	23.6	Technical	M	Brandon Bray	Write up array initialization.		No	

	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
79	16-Dec-03	Phone meeting	27	Technical	H	Brandon Bray	Cover unification of CLI and Standard C++ exception-handling models, and anything else that might go in this clause.  Are exceptions asynchronous now in some cases? Yes they are. (For example, <code>NullReferenceException</code> .)	Meeting #5 (WA): Kevin Free (Microsoft) gave a verbal presentation.  catch(...) catches managed and native exceptions.  catch(System::Object^) also catches both kinds, but won't invoke the destructor (so can leak).  CLI exception handling supports more features than we expose.  The issue remained with Brandon to write up, as before.	No	
82	16-Dec-03	Phone meeting	29	Technical	M	Brandon Bray	Flesh out "Templates" clause.		No	
94	29-Jan-04	meeting #2 (HI)		Technical		Mark Hall	Relationship between primitive types and CLI types.  The current spec allows the following: <code>int i = 10; String^ s = i.ToString();</code> Standard C++ doesn't allow member selection on expressions of primitive type. Assuming <code>int</code> maps to <code>System::Int32</code> , just how much alike are these two types? Specifically, when do we treat the primitive as the underlying class.	Meeting 5 (WA): Asked Mark to write this up and distribute to the reflector. Please address the side-effect issue; that is, given <code>(i++)</code> .ToString(), is the increment done?  Meeting 7 (WA): ?? To be done in Tue morning work sessions.  Re the side-effect, yes, it must be done.	No	
98	29-Jan-04	meeting #2 (HI)	30	Technical	R	Brandon Bray	Restrictions on generics re generic code generation.  The current generics clause needs to be fleshed out, especially w.r.t how overload resolution works within the CLI.	Meeting #2 (HI): Brandon will write a paper on this.  Meeting #4 (NJ): The fleshing out of Clause 30 is a significant contribution toward this. More work needed in declarations and function calls.	No	
105	29-Jan-04	meeting #2 (HI)	14.5.1	Technical		Mark Hall	Constructors can't be used in casts in managed classes. Should they be allowed in explicit conversions? All managed type constructors being explicit by default. (Already yes, but reconfirm this.)	Meeting #4 (NJ): Steve will send the editor sufficient text to go into the public drop to indicate our intention re this topic. DONE.  Meeting 5 (WA): Asked Mark to write this up and distribute to the reflector.  Meeting 7 (WA): Steve and Mark worked on this and presented it to the full committee on the 2nd day. Mark will write this up for future consideration.	No	
111	19-Feb-04		15.3.2	Technical	M	Brandon Bray	Need to consider how indexed access expressions are interpreted in templates.		No	
117	19-Feb-04		18.4.2	Technical	H	Brandon Bray	The qualified name of a property needs to be described somewhere. Once that happens, how an out-of-class definition is done will already be covered by existing rules.		No	

	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
124	10-Jun-04	Jonathan Caves		Technical		Jonathan Caves	<p>Indexed properties -- Consider the following:</p> <pre> interface class I1 {   property int Value; };  interface class I2 {   property int Value[String^] {     int get(String^);     void set(String^, int);   }; };  ref class D : I1, I2 {   // Implements the properties };  D^ d; d-&gt;Value["Foo"];  The question is what does the last line do?  Which leads to a language design question - what should the compiler do when faced with a property followed by a '['  1) Should it look for just parameterized properties and if there isn't one fail - I suspect not  2) Should it look for all properties and if the returned set contains a parameterized property it should prefer it - this sounds like magic to me.  3) Should it look for all properties perform overload resolution across the whole set and if the resulting call is ambiguous then issue an error.  Mark Hall says: Jonathan's looking into deferring the property tree rewrite until we know everything about </pre>	<p>Meeting #5 (WA): Discussed this. Option #3 preferred.</p> <p>Meeting 7 (WA): Discussed this in detail.</p> <pre> property int Value[int] {   void set(int, int); };  x-&gt;Value[1] = 4 is treated as x-&gt;set_Value(1,4);  -----  property array&lt;int&gt;^ Value {   array&lt;int&gt;^ get(); }  x-&gt;Value[1] = 4 is treated as x-&gt;get_Value()[1] = 4  -----  property int% Value[int] {   int% get(int); }  x-&gt;Value[1] = 4 is treated as x-&gt;get_Value(1) = 4  This construct violates the principle of properties (that of setting/getting the value of some property), so is not to be encouraged: however, it is supported, but no need to consider it further here.  ----- </pre>	No	
125	14-Jun-04	meeting #5 (WA)	8.15.3	Technical	M	Brandon Bray	<p>Based on the rules for type deduction in templates, it seems surprising that you can match array&lt;ItemType&gt;^ with an argument of type int. Here is a standard C++ example intended to illustrate the issue:</p> <pre> template &lt;class ItemType&gt; struct Stack {}; template &lt;class ItemType&gt; struct Array {   Array(ItemType); }; template &lt;class ItemType&gt; void PushMultiple(Stack&lt;ItemType&gt;, Array&lt;ItemType&gt;); int main() {   Stack&lt;int&gt; s;   PushMultiple(s, 1); // deduction fails   PushMultiple&lt;int&gt;(s, 1); }  Are the rules for generic different in this area? [There seems to be information related to this in 30.3.2. See that subclause for further comments on this issue.] </pre>		No	

	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
132	14-Jun-04	meeting #5 (WA)	15.3.10	Technical	M	Brandon Bray	Unboxing and boxing are described as preferred user-defined conversions; however, this is incorrect.		No	
138	14-Jun-04	meeting #5 (WA)	18.4	Technical		Mark Hall	Need to write up the restrictions on trivial properties.		No	
143	14-Jun-04	meeting #5 (WA)	19.7	Technical	L	Brandon Bray	The restriction below does not apply to non-static member operators – that need not have a parameter of the type of the class.	Meeting #9 (NJ): Needs to be done.	No	
151	14-Jun-04	meeting #5 (WA)	25.2	Technical	M	Brandon Bray	The note says "pickup the restrictions from page 333 (of Brandon's paperback copy of the C# spec)".		No	
153	14-Jun-04	meeting #5 (WA)	30.1	Technical	M	Brandon Bray	The text indicates that a generic-declaration may appear in a class scope, but the syntax of member-declaration has not been extended to permit a generic-declaration. [[#98]]		No	
154	14-Jun-04	meeting #5 (WA)	30.1	Technical	R	Brandon Bray	Doesn't the text "a generic name declared in namespace scope or in class scope shall be unique in that scope" make the first sentence of this paragraph redundant? Re the reference to 14.5.4: That is the section on partial specialization. Generics can't be partially specialized, can they? The spec. should probably answer that explicitly.		No	
155	14-Jun-04	meeting #5 (WA)	30.1	Technical	R	Brandon Bray	What is a non-generic type? Does it mean that the rules are the same as classes? As template classes? Something else?		No	
158	14-Jun-04	meeting #5 (WA)	30.1.1	Technical	R	Brandon Bray	The equivalent wording for template parameters in the working paper has been changed to "defines its identifier to be a typedef-name". The revised wording should probably be used here too (see core issue 283)		No	
160	14-Jun-04	meeting #5 (WA)	30.1.6	Technical	R	Brandon Bray	This subclause describes when a static constructor is invoked. In 18.8, it references the CLI Standard Partition II (10.5.3). Are the rules the same? (Yes) Should this subclause also just reference the CLI spec? There are two sets of behavior; we need to say which one we use.		No	
161	14-Jun-04	meeting #5 (WA)	30.1.7	Technical	M	Brandon Bray	What to say about explicit conversion functions (which can only occur in managed class types)?		No	
162	14-Jun-04	meeting #5 (WA)	30.2.2	Technical	R	Brandon Bray	This subclause lists the types that can and cannot be generic arguments. Fundamental types are not included in either set, neither are function types. The subclause does not say whether or not cv-qualified types are allowed.		No	

	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
167	14-Jun-04	meeting #5 (WA)	30.3	Technical	L	Brandon Bray	<p>"When the type of a parameter or variable is a type parameter, the declaration of that parameter or variable shall use that type parameter's name without any pointer, reference, or handle declarators."</p> <p>What about cv-qualifiers?</p>	Meeting #9 (NJ): Needs to be done. CV-qualifiers are not permitted.	No	
169	14-Jun-04	meeting #5 (WA)	30.3.2	Technical	L	Brandon Bray	<p>The issue raised in 8.15.3 is somewhat answered here. 18.3.6 seems to deal with expanded forms of calls, not expanded forms of function declarations. I interpret the text above as saying that deduction is done as if the function were declared like this:</p> <pre>generic &lt;typename ItemType&gt; void PushMultiple(Stack&lt;ItemType&gt;^, ItemType i1, ItemType i2,/* ... */);</pre> <p>Is that correct? I think this requires a more detailed description.</p>	Meeting #9 (NJ): Needs to be done. Add example(s).	No	
170	14-Jun-04	meeting #5 (WA)	30.3.2	Technical	L	Brandon Bray	Something needs to be said about instantiating a generic delegate using a generic function.	Meeting #9 (NJ): Needs to be done.	No	
173	14-Jun-04	meeting #5 (WA)	32.1.4	Technical	L	Brandon Bray	To ensure that signatures for the same Type produced by different implementations match, the ordering in such a set of modreqs and modopts is as follows: first modreqs in ascending order by name, then modopts in ascending order by name, with case being significant. [[We need some rule here; is this the one?]].	Meeting #9 (NJ): Add a description of our best guess at the correct solution, to Future Directions, then mark this Postponed. Point to this from the normative text somehow.	No	

	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
184	2-Aug-04	meeting #6 (WA)		Technical		Herb Sutter	Describe problem with overloading on % vs. &  Herb presented the following code:  <pre>#include &lt;iostream&gt; using namespace std; void f( const int&amp; ) { cout &lt;&lt; "f( const int&amp; )" &lt;&lt; endl; } void f( int&amp; )      { cout &lt;&lt; "f( int&amp; )" &lt;&lt; endl; }  void g( int% )      { cout &lt;&lt; "g( int% )" &lt;&lt; endl; } void g( int&amp; )      { cout &lt;&lt; "g( int&amp; )" &lt;&lt; endl; }  int main() {     const int ci = 0;     int i = 0;     int^ hi = gcnew int;      f( ci );     f( i );      g( *hi );     // g( i );    // ambiguous: should g(int&amp;) be                 // preferred? }  The following code was his attempt to write an agnostic swap:  template&lt;typename T&gt; void swap( T% a, T% b ) {     #if defined NO_PIN_PTR          // doesn't work         T temp = a; a = b; b = temp;     #elif defined PIN_PTR_BUG      // doesn't         compile         T temp = *pin_ptr&lt;T&gt;(a);         *pin_ptr&lt;T&gt;(*pa) = *pin_ptr&lt;T&gt;(*pb);         *pin_ptr&lt;T&gt;(*pb) = temp;     #else                          // does compile -- but</pre>		No	
194	2-Aug-04	Anthony Williams	15.3.2	Technical		Jonathan Caves	Re Anthony's post to the reflector re "default index	Meeting 7 (WA): Discussed the possibility of disallowing both the default indexed property and operator[].	No	
196	30-Sep-04	meeting #7 (WA)		Technical		Herb Sutter	In native types, % behaves like &.		No	
198	30-Sep-04	meeting #7 (WA)	2	Technical		Herb Sutter	Propose wording to require that extensions over and above ISO C++ requirements, be diagnosed.	Meeting 9 (NJ): Re the new paragraph added to §2. "Conformance" in response to spreadsheet issue #198, the committee believed this text does not adequately address the issue. The editor was asked to remove it.  Ownership was transferred from Tom to Herb.	No	
203	26-Oct-04	Rex Jaeschke	10.1.2	Technical	M	Brandon Bray	[Note: The compiler needs to add typedef members to the class so that template code can use the return type or the parameter types. [[Need more explanation.]] end note]		No	
204	26-Oct-04	Rex Jaeschke	12.2.2	Technical	M	Brandon Bray	Write intro text.		No	

	A	B	C	D	E	F	G	H	I	J
1	Date Raised?	Issue Raiser?	Reference	Issue Type	Priority	Owner	Comment	Other Remarks	Resolved?	Postponed?
205	26-Oct-04	Rex Jaeschke	15.5	Technical	H	Brandon Bray	<p>15.5 Explicit type conversion (cast notation)</p> <p>The rules in the C++ Standard (§5.4/5) have been extended for C++/CLI by including safe casts before static casts.</p> <ul style="list-style-type: none"> <li>• a const_cast</li> <li>• a safe_cast</li> <li>• a safe_cast followed by a const_cast</li> <li>• a static_cast</li> <li>• a static_cast followed by a const_cast</li> <li>• a reinterpret_cast</li> <li>• a reinterpret_cast followed by a const_cast</li> </ul> <p>[Note: Standard C++ programs remain unchanged by this, as safe casts are ill-formed when either the expression type or target type is a native class. end note]</p> <p>Provide background on the expected behavior and rationale. (Get this from the updated casting proposal.)</p>		No	
206	26-Oct-04	Rex Jaeschke	21.4	Technical	M	Brandon Bray	Simple value classes: Flesh this out.		No	
207	26-Oct-04	Rex Jaeschke	24.2.5	Technical	H	Brandon Bray	Interface member access: Write up.		No	
210	4-Dec-04	Rex Jaeschke	29.5.1	Technical	M	Brandon Bray	There is confusion about DefaultMember attribute and IndexNameAttribute. In the current implementation, it appears that the first one is exhibiting the behavior of the second one, and the second one is being emitted into metadata directly when it should be consumed by the compiler.		No	
213	4-Dec-04	Rex Jaeschke		Technical	M	Brandon Bray	13.3.3.2/4 of the C++ Standard has rules for pointer conversions, that need to be adapted to handles. Review this subclause and determine the changes needed for the C++/CLI spec.		No	
218	Feb-10-2005	Rex Jaeschke	15.3	Technical	M	Brandon Bray	<p>Are the productions</p> <pre> postfix-expression . pseudo-finalizer-name postfix-expression -&gt; pseudo-finalizer-name </pre> <p>necessary, and, if so, should the "pseudo-" prefix be dropped?</p>		No	
219	3-Mar-05	Brandon Bray		Technical	M	Brandon Bray	Currently, the Visual C++ compiler allows a friend to first declare a generic type. Whether the language specification says this is allowed is up for discussion. Are there any issues we should consider before saying that it should be supported?		No	



[illegible]

**Minutes of the:  
held in:  
on:**

**10<sup>th</sup> meeting of Ecma TC39-TG5  
Kamuela, HI, USA  
7 and 8 March 2005**

Rex Jaeschke

[rex@RexJaeschke.com](mailto:rex@RexJaeschke.com)

2005-03-08

## **1 Opening**

Convener Tom Plum welcomed everyone to the tenth meeting of TG5.

### **1.1 Appointment of Recording Secretary**

Rex Jaeschke was appointed.

### **1.2 Introduction of participants**

The participants introduced themselves. Those attending were: Sean Perry (IBM), Brandon Bray (Microsoft), Rex Jaeschke (Microsoft), P.J. Plauger (Dinkumware), Tana Plauger (Dinkumware), Tom Plum (Plum Hall), Herb Sutter (Microsoft), and Toshiaki Kurokawa (CSK Corp.).

### **1.3 Host facilities/local information**

Local information was provided.

## **2 Adoption of the agenda**

Document 2005-08 was approved without objection. Under “Other Business”, we’ll cover Brandon’s recent emails.

## **3 Approval of Minutes of previous TG5 meeting**

Document 2005-07 was approved without objection.

## **4 Matters arising from the minutes not covered elsewhere**

None.

## **5 Project Editor’s Report – Rex Jaeschke**

Rex presented document 2005-11.

## **6 Approving tracked changes in latest draft**

During our discussion of WD1.10 (2005-09), numerous issues were raised, and the editor took copious notes for the next draft.

WD1.10 was adopted as the new base document with the usual caveat that it still needs quite a bit of work.

## 7 Date and place of next meetings

### 7.1 Next Meeting

**Sep 19-20, 2005:** Redmond, WA; hosted by Microsoft  
(and, if needed, 1 hour at 9 am on Sep 23 to review work done post-main meeting)

(The TC39 business meeting will be Sep 23, 1-5 pm.)

*Action:* Microsoft will host a teleconference facility for the Sep meeting; Herb will distribute the dial-in information.

The following phone conferences were scheduled to occur at 10 AM PST, for 2 hours max:  
May 5, Jun 9, Jun 23, Jul 7, Jul 21, Aug 4, Aug 18.

*Action:* Microsoft will host the teleconference facility; Herb will distribute the dial-in information.

## 8 Reports from Liaisons

### 8.1 TC39 TG3 (CLI) – Rex Jaeschke

CLS Rule 25 has been withdrawn. This rule previously stated “The accessibility of a property’s accessors shall be identical.”

### 8.2 SC22/WG21 (C++) – Tom Plum, P.J. Plauger, Tana Plauger, John Spicer, and Steve Adamczyk.

No meeting has been held since the last TG5 meeting.

### 8.3 TC39 TG2 (C#) – Rex Jaeschke

As Tom reported in a recent phone meeting, TG2 agreed to rename “destructor” to “finalizer”.

### 8.4 ISO/IEC JTC 1/SC 22 – Rex Jaeschke

None.

## 9 Action item and comment spreadsheet review

### 9.1 “Issue 172” email from Brandon Bray 2005-03-03.

Item 1 (special constraints):

*Action:* Editor will forward to Brandon the new text and table from the CLI spec Partition II w.r.t special constraints.

Item 6 (friends):

*Action:* Editor will notify John Spicer (who originated these questions) that TG5 decided that a generic can first be declared in a friend declaration.

Once Brandon’s proposal has been applied to the draft, this issue will be closed.

Some of the text rewrite done for this issue also allows #171 to be closed.

### 9.2 “Issue 76” from Brandon Bray 2005-03-03.

Once Brandon’s proposal has been applied to the draft, this issue will be closed.

### 9.3 “Issue 215” from Brandon Bray 2005-03-03.

It was agreed that the replacement words go into a different place; in the middle of the 2nd paragraph of §12.3.5.

The aim is to have proposals that resolve the Priority H issues in time for the May 5 phone call.

**9.4 “FW: Review of Chapters 15-23” from Jonathan Caves (via Brandon Bray) 2005-03-07, and**

**9.5 “Dinkumware notes” from P.J. Plauger 2005-03-08**

These comments were at least partially reviewed.

*Action:* Editor will draft responses to these issues, and will post them to the liaison reflector.

## **10 Approval of TG5 spec to forward to TC39**

It was agreed that the draft is not ready to be forwarded at this time.

## **11 Any other business**

### **11.1 Distribution of docs to WG21:**

*Action:* Editor will distribute to the TG5 reflector, WD1.10, so members can make it available on their websites for access by WG21 members. Editor will also announce this availability to the liaison email reflector.

*Action:* Editor will concatenate the PDFs of all docs (except WD1.10) to WG21, and forward to Herb for distribution. (This package will include these draft minutes after TG5 has had a change to review and correct them via email.) This packet will include a document containing URLs from which the latest draft can be obtained.

### **11.2 Thank meeting host:**

Everyone thanked meeting host Plum Hall for meeting and catering arrangements.

## **12 Adjournment**

The meeting was adjourned mid-afternoon March 8.