

Activity Streams Working Group

J. Snell
IBM
M. Atkins
SAY Media
W. Norris
Google
C. Messina
Citizen Agency, Google
M. Wilkinson
MySpace, Facebook, VMware
R. Dolin
Microsoft
May 2011

TOC

JSON Activity Streams 1.0

Abstract

This specification details the serialization of a stream of social activities using the JSON format. Activities are important in that they allow individuals to process the latest news of people and things they care about.

Table of Contents

- [1. Introduction](#)**
- [2. Notational Conventions](#)**
- [3. JSON Serialization](#)**
 - [3.1. Example Activities](#)**
 - [3.2. Activity Serialization](#)**
 - [3.3. Media Link Serialization](#)**
 - [3.4. Object Serialization](#)**
 - [3.5. Collection Serialization](#)**
- [4. The Activity Stream JSON Document](#)**
- [5. Comparison of Identifier Values](#)**
- [6. The "post" Verb](#)**
 - [6.1. Implied "post" Activity](#)**
- [7. The "activity" Object Type](#)**
- [8. Extension Object Types and Verbs](#)**
- [9. Other Representations](#)**
- [10. Requirements for Re-publishers](#)**
- [11. Security Considerations](#)**
- [12. IANA Considerations](#)**
- [13. License](#)**
- [14. Errata](#)**
- [15. Normative References](#)**
- [Appendix A. Acknowledgements](#)**
- [§ Authors' Addresses](#)**

TOC

1. Introduction

In its simplest form, an `activity` consists of an `actor`, a `verb`, an `object`, and a `target`. It tells the story of a person performing an action on or with an object -- "Geraldine posted a photo to her album" or "John shared a video". In most cases these components will be explicit, but they may also be implied.

It is a goal of this specification to provide sufficient metadata about an activity such that a consumer of the data can present it to a user in a rich human-friendly format. This may include constructing readable sentences about the activity that occurred, visual representations of the activity, or combining similar activities for display.

The basic properties that comprise the description of an `activity` are defined in the following sections.

Within this specification, an `object` is a thing, real or imaginary, which participates in an activity. It may be the entity performing the activity, or the entity on which the activity was performed. An `object` consists of properties defined in the following sections. Certain object types may further refine the meaning of these properties, or they may define additional properties.

Some types of objects may have an alternative visual representation in the form of an image, video or embedded HTML fragments. A `Media Link` represents a hyperlink to such resources.

An `Activity Stream` is a collection one or more individual activities. The relationship between the activities within the collection is undefined by this specification.

2. Notational Conventions

TOC

The text of this specification provides the sole definition of conformance. Examples in this specification are non-normative.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [\[RFC2119\]](#).

This specification allows the use of IRIs [\[RFC3987\]](#). Every URI [\[RFC3986\]](#) is also an IRI, so a URI may be used wherever below an IRI is named. There are two special considerations: (1) when an IRI that is not also a URI is given for dereferencing, it MUST be mapped to a URI using the steps in Section 3.1 of [\[RFC3987\]](#) and (2) when an IRI is serving as an `id` value, it MUST NOT be so mapped.

3. JSON Serialization

TOC

Activities are serialized using the JSON format, as defined in [\[RFC4627\]](#). Alternative serializations MAY be used but are outside the scope of this specification.

In the JSON serialization, absent properties MAY be represented either by an explicit declaration of the property whose value is null or by omitting the property declaration altogether at the option of the publisher; these two representations are semantically equivalent. If a property is having a value whose type is a JSON array, the absence of any items in that array MUST be represented by omitting the property entirely or publishing it with the value null, and MUST NOT be represented as an empty array, except as otherwise stated in the definition of a specific property.

Unless otherwise specified, all properties specifying date and time values within the JSON

serialization, including extensions, MUST conform to the "date-time" production in [\[RFC3339\]](#). In addition, an uppercase "T" character MUST be used to separate date and time, and an uppercase "Z" character MUST be present in the absence of a numeric time zone offset.

[TOC](#)

3.1. Example Activities

Following is a simple, minimal example of a JSON serialized [activity](#):

```
{
  "published": "2011-02-10T15:04:55Z",
  "actor": {
    "url": "http://example.org/martin",
    "objectType": "person",
    "id": "tag:example.org,2011:martin",
    "image": {
      "url": "http://example.org/martin/image",
      "width": 250,
      "height": 250
    },
    "displayName": "Martin Smith"
  },
  "verb": "post",
  "object": {
    "url": "http://example.org/blog/2011/02/entry",
    "id": "tag:example.org,2011:abc123/xyz"
  },
  "target": {
    "url": "http://example.org/blog/",
    "objectType": "blog",
    "id": "tag:example.org,2011:abc123",
    "displayName": "Martin's Blog"
  }
}
```

A more extensive, single-entry [Activity Stream](#) follows. In addition to containing a number of required and optional core properties, the example contains the additional, undefined extension properties "foo" and "foo2" for illustrative purposes only.

```
{
  "items": [
    {
      "published": "2011-02-10T15:04:55Z",
      "foo": "some extension property",
      "generator": {
        "url": "http://example.org/activities-app"
      },
      "provider": {
        "url": "http://example.org/activity-stream"
      },
      "title": "Martin posted a new video to his album.",
      "actor": {
        "url": "http://example.org/martin",
        "objectType": "person",
        "id": "tag:example.org,2011:martin",
        "foo2": "some other extension property",
        "image": {
          "url": "http://example.org/martin/image",

```

```

        "width": 250,
        "height": 250
    },
    "displayName": "Martin Smith"
},
"verb": "post",
"object": {
    "url": "http://example.org/album/my_fluffy_cat.jpg",
    "objectType": "photo",
    "id": "tag:example.org,2011:my_fluffy_cat",
    "image": {
        "url": "http://example.org/album/my_fluffy_cat_thumb.jpg",
        "width": 250,
        "height": 250
    }
},
"target": {
    "url": "http://example.org/album/",
    "objectType": "photo-album",
    "id": "tag:example.org,2011:abc123",
    "displayName": "Martin's Photo Album",
    "image": {
        "url": "http://example.org/album/thumbnail.jpg",
        "width": 250,
        "height": 250
    }
}
}
}
]
}

```

3.2. Activity Serialization

TOC

Property	Value	Description
<code>actor</code>	Object	Describes the entity that performed the activity. An activity MUST contain one <code>actor</code> property whose value is a single Object .
<code>content</code>	JSON [RFC4627] String	Natural-language description of the activity encoded as a single JSON String containing HTML markup. Visual elements such as thumbnail images MAY be included. An activity MAY contain a <code>content</code> property.
<code>generator</code>	Object	Describes the application that generated the activity. An activity MAY contain a <code>generator</code> property whose value is a single Object .
<code>icon</code>	Media Link	Description of a resource providing a visual representation of the object, intended for human consumption. The image SHOULD have an aspect ratio of one (horizontal) to one (vertical) and SHOULD be suitable for presentation at a small size. An activity MAY have an <code>icon</code> property.
<code>id</code>	JSON [RFC4627] String	Provides a permanent, universally unique identifier for the activity in the form of an absolute IRI [RFC3987] . An activity SHOULD contain a single <code>id</code> property. If an activity does not contain an <code>id</code> property, consumers MAY use the value of the <code>url</code> property as a less-reliable, non-unique identifier.
<code>object</code>	Object	Describes the primary object of the activity. For instance, in the activity, "John saved a movie to his wishlist", the <code>object</code> of the activity is "movie". An activity SHOULD contain an <code>object</code>

		property whose value is a single Object . If the <code>object</code> property is not contained, the primary object of the activity MAY be implied by context.
<code>published</code>	[RFC3339] date-time	The date and time at which the activity was published. An activity MUST contain a <code>published</code> property.
<code>provider</code>	Object	Describes the application that published the activity. Note that this is not necessarily the same entity that generated the activity. An activity MAY contain a <code>provider</code> property whose value is a single Object .
<code>target</code>	Object	Describes the target of the activity. The precise meaning of the activity's target is dependent on the activities <code>verb</code> , but will often be the object the English preposition "to". For instance, in the activity, "John saved a movie to his wishlist", the target of the activity is "wishlist". The activity target MUST NOT be used to identity an indirect object that is not a target of the activity. An activity MAY contain a <code>target</code> property whose value is a single Object .
<code>title</code>	JSON [RFC4627] String	Natural-language title or headline for the activity encoded as a single JSON String containing HTML markup. An activity MAY contain a <code>title</code> property.
<code>updated</code>	[RFC3339] date-time	The date and time at which a previously published activity has been modified. An Activity MAY contain an <code>updated</code> property.
<code>url</code>	JSON [RFC4627] String	An IRI [RFC3987] identifying a resource providing an HTML representation of the activity. An activity MAY contain a <code>url</code> property.
<code>verb</code>	JSON [RFC4627] String	Identifies the action that the activity describes. An activity SHOULD contain a <code>verb</code> property whose value is a JSON String that is non-empty and matches either the "isegment-nz-nc" or the "IRI" production in [RFC3339] . Note that the use of a relative reference other than a simple name is not allowed. If the <code>verb</code> is not specified, or if the value is null, the <code>verb</code> is assumed to be "post" .

3.3. Media Link Serialization

TOC

Property	Value	Description
<code>duration</code>	JSON [RFC4627] int	A hint to the consumer about the length, in seconds, of the media resource identified by the <code>url</code> property. A media link MAY contain a "duration" property when the target resource is a time-based media item such as an audio or video.
<code>height</code>	JSON [RFC4627] int	A hint to the consumer about the height, in pixels, of the media resource identified by the <code>url</code> property. A media link MAY contain a <code>height</code> property when the target resource is a visual media item such as an image, video or embeddable HTML page.
<code>url</code>	String	The IRI of the media resource being linked. A media link MUST have a <code>url</code> property.
<code>width</code>	JSON [RFC4627] int	A hint to the consumer about the width, in pixels, of the media resource identified by the <code>url</code> property. A media link MAY contain a <code>width</code> property when the target resource is a visual media item such as an image, video or embeddable HTML page.

3.4. Object Serialization

Property	Value	Description
<code>attachments</code>	JSON [RFC4627] Array of Objects	A collection of one or more additional, associated objects, similar to the concept of attached files in an email message. An object MAY have an <code>attachments</code> property whose value is a JSON Array of Objects .
<code>author</code>	Object	Describes the entity that created or authored the object. An object MAY contain a single <code>author</code> property whose value is an Object of any type. Note that the author field identifies the entity that created the object and does not necessarily identify the entity that published the object. For instance, it may be the case that an object created by one person is posted and published to a system by an entirely different entity.
<code>content</code>	JSON [RFC4627] String	Natural-language description of the object encoded as a single JSON String containing HTML markup. Visual elements such as thumbnail images MAY be included. An object MAY contain a <code>content</code> property.
<code>displayName</code>	JSON [RFC4627] String	A natural-language, human-readable and plain-text name for the object. HTML markup MUST NOT be included. An object MAY contain a <code>displayName</code> property. If the object does not specify an <code>objectType</code> property, the object SHOULD specify a <code>displayName</code> .
<code>downstreamDuplicates</code>	JSON [RFC4627] Array of Strings	A JSON Array of one or more absolute IRI's [RFC3987] identifying objects that duplicate this object's content. An object SHOULD contain a <code>downstreamDuplicates</code> property when there are known objects, possibly in a different system, that duplicate the content in this object. This MAY be used as a hint for consumers to use when resolving duplicates between objects received from different sources.
<code>id</code>	JSON [RFC4627] String	Provides a permanent, universally unique identifier for the object in the form of an absolute IRI [RFC3987] . An object SHOULD contain a single <code>id</code> property. If an object does not contain an <code>id</code> property, consumers MAY use the value of the <code>url</code> property as a less-reliable, non-unique identifier.
<code>image</code>	Media Link	Description of a resource providing a visual representation of the object, intended for human consumption. An object MAY contain an <code>image</code> property whose value is a Media Link .
<code>objectType</code>	JSON [RFC4627] String	Identifies the type of object. An object MAY contain an <code>objectType</code> property whose value is a JSON String that is non-empty and matches either the "isegment-nz-nc" or the "IRI" production in [RFC3987] . Note that the use of a relative reference other than a simple name is not allowed. If no <code>objectType</code> property is contained, the object has no specific type.
<code>published</code>	[RFC3339] date-time	The date and time at which the object was published. An object MAY contain a <code>published</code> property.

<code>summary</code>	JSON [RFC4627] String	Natural-language summarization of the object encoded as a single JSON String containing HTML markup. Visual elements such as thumbnail images MAY be included. An activity MAY contain a <code>summary</code> property.
<code>updated</code>	[RFC3339] date-time	The date and time at which a previously published object has been modified. An Object MAY contain an <code>updated</code> property.
<code>upstreamDuplicates</code>	JSON [RFC4627] Array of Strings	A JSON Array of one or more absolute IRI's [RFC3987] identifying objects that duplicate this object's content. An object SHOULD contain an <code>upstreamDuplicates</code> property when a publisher is knowingly duplicating with a new ID the content from another object. This MAY be used as a hint for consumers to use when resolving duplicates between objects received from different sources.
<code>url</code>	JSON [RFC4627] String	An IRI [RFC3987] identifying a resource providing an HTML representation of the object. An object MAY contain a <code>url</code> property

3.5. Collection Serialization

TOC

A "collection" is a generic list of **Objects** of any object type. The `objectType` of each item in the collection MAY be omitted if the type of object can be established through context. The collection is used primarily as the root of an Activity Streams document as described in **Section 4**, but can be used as the value of extension properties in a variety of situations.

Property	Value	Description
<code>totalItems</code>	JSON [RFC4627] Integer	Non-negative integer specifying the total number of activities within the stream. The Stream serialization MAY contain a <code>count</code> property.
<code>items</code>	JSON [RFC4627] Array of Objects	An array containing a listing of Objects of any object type. If used in combination with the <code>url</code> property, the <code>items</code> array can be used to provide a subset of the objects that may be found in the resource identified by the <code>url</code> .
<code>url</code>	JSON [RFC4627] String	An IRI [RFC3987] referencing a JSON document containing the full listing of objects in the collection.

A valid Collection object serialization MUST contain at least the `url` or `items` properties.

4. The Activity Stream JSON Document

TOC

The above defined JSON serialization can be used to represent activities, objects and media links in any JSON context. This section defines one particular use of the above formats to publish a JSON document representing a stream of activities.

Publishers using this format MUST produce a valid JSON document whose root value is a **Collection**. This object MUST, at a minimum, include the `items` property whose objects MUST be **Activities**.

5. Comparison of Identifier Values

TOC

The values of multiple **Object's** `id`, `upstreamDuplicates`, and `downstreamDuplicates` properties can be compared to determine if the objects represent duplicate content. Processors **MUST** compare these values on a character-by-character basis (in a case-sensitive fashion). Comparison operations **MUST** be based solely on the IRI character strings and **MUST NOT** rely on dereferencing the IRIs or URIs mapped from them.

As a result, two IRIs that resolve to the same resource but are not character-for-character identical will be considered different for the purposes of identifier comparison.

6. The "post" Verb

TOC

This specification defines one initial verb and defers to other specifications to define additional verb and object types as defined in **Section 8**.

The "post" verb describes the act of authoring an object and then publishing it online. The `actor` can be any entity; the `object` can be of any object type; and the `target`, if specified, can be of any object type. A `target`, however, is not required.

6.1. Implied "post" Activity

TOC

Because the "post" verb is defined to include the act of publishing an object online, it is possible to assume an implied "post" activity for any object even if such an activity was never explicitly published or received. The properties of this implied activity are as follows; any property not explicitly listed here is absent from the resulting activity unless the consumer is able to infer a value based on out-of-band information outside the scope of this specification:

actor

The `actor` of the implied activity is unknown unless the consumer is able to infer an `actor` based on information outside the scope of this specification. For example, the consumer may know via some out-of-band information that all of the objects in a particular document were posted by a particular user.

object

The `object` of the implied activity is the object for which this implied activity is being constructed.

verb

The `verb` of the implied activity is "post".

published

The `published` of the implied activity is unknown unless the consumer is able to infer a time based on information outside the scope of this specification.

7. The "activity" Object Type

TOC

The "activity" object type indicates that an Object represents an Activity. It can be used whenever an event involving another activity is described within an Activity Stream. For instance, when one user "shares" or "reposts" an activity from another user.

In addition to the core Object properties, an "activity" object **MAY** contain any combination of the properties defined for an **Activity**. The values of the object's `id` and `url` fields **MUST** match those of the activity being described.

Note that an `activity` object MUST be processable as both an **Object** and an **Activity** and may contain fields from either construct, some of which may overlap or duplicate content. When duplication of content may occur, an implementation SHOULD choose to use the fields from the **Activity** object. For instance, in an `activity` object, the Activity's `title` property and the Object's `displayName` properties serve identical functions, therefore the use of the `title` property SHOULD be favored over the use of `displayName`.

The example below illustrates an activity in which one user, Jane, is sharing another activity performed by a second user, John.

```
{
  "published": "2011-02-10T15:04:55Z",
  "actor": {
    "objectType": "person",
    "id": "tag:example.org,2011:jane"
  },
  "verb": "share",
  "object": {
    "objectType": "activity",
    "title": "John posted a photo",
    "id": "tag:example.org,2011:abc123",
    "verb": "post",
    "actor": {
      "objectType": "person",
      "id": "tag:example.org,2011:john"
    },
    "object": {
      "objectType": "photo",
      "url": "http://example.org/album/my_fluffy_cat.jpg"
    }
  }
}
```

TOC

8. Extension Object Types and Verbs

Other specifications MAY define new object types and verbs for use with the concepts and serializations defined in this specification. To be clear, new extension properties can be added anywhere in the JSON serialization of an object or activity.

When defining a new verb, the defining specification MUST provide a verb identifier for the new verb and MUST define the meaning of the verb in terms of the `actor`, `object` and `target` properties of the activity, and MUST define whether a `target` is required and what it represents when present.

When defining a new object type, the defining specification MUST provide an object type identifier for the new object type. The definition of the new object type MAY define additional properties specific to that object type; if so, the specification SHOULD define how these additional properties are to be serialized within the JSON document, and the specification MAY define how these additional properties are to be represented in other serialization formats. The definition of the new object type MAY impose additional meaning or constraint on the core object properties as long as the new definition remains compatible with the definition of the base component.

Additional extension properties MAY also be defined for **Activities** and **Media Links**. The specification of these properties SHOULD define how they are to be serialized within the JSON document, and the specification MAY define how they are to be represented in other serialization formats.

New properties SHOULD be defined such that the **Object**, **Activity** or **Media Link** would remain useful if those properties are omitted or removed during processing, to enable graceful fallback by consumers that do not implement the extension properties or alternative representations that are unable to represent the extensions.

Processors that encounter unfamiliar extension properties within an **Object**, **Activity** or **Media Link** MUST NOT stop processing or signal an error and MUST continue processing the items as if the extension properties were not present.

TOC

9. Other Representations

This specification defines the primary representation for "activities", "objects" and "media links" in terms of the JSON serialization format. Other specifications MAY define additional serializations of these items. If the new serialization is compatible with the JSON data model, the new representation SHOULD be defined as a transformation of the raw JSON structures.

Defining a new representation as a transformation from JSON will allow the new representation to automatically support additional properties added for new object types as described in **Section 8**. Otherwise the representation of these new properties must be defined on a case-by-case basis.

TOC

10. Requirements for Re-publishers

A Re-publisher is an agent which receives "activities" or "objects" and then transmits these items to other agents.

When a Re-publisher transmits an "activity", the Re-publisher MUST preserve the original `verb`, `published`, `actor`, `object` and `target` properties. The Re-publisher MAY preserve the remaining properties, OR the Re-publisher MAY omit these properties entirely or replace these properties with locally-determined values.

When a Re-publisher transmits an `object`, the Re-publisher MUST maintain the full integrity of the `object`, including any extension properties, and retain the original `id` value OR, if maintaining full fidelity is not possible the Re-publisher MUST assign this copy a new `id` and include the `id` of the original object as a value of the `upstreamDuplicates` property of the new copy. In the latter case, a new `object` is created which is considered to be a duplicate of the original.

TOC

11. Security Considerations

Publishers or Consumers implementing Activity Streams as a stream of public data may also want to consider the potential for unsolicited commercial or malicious content and should take preventative measures to recognize such content and either identify it or not include it in their stream implementations.

Publishers should take reasonable measures to make sure potentially malicious user input such as cross-site scripting attacks are not included in the Activity Streams data they publish.

Consumers that re-emit ingested content to end-users MUST take reasonable measures if emitting ingested content to make sure potentially malicious ingested input is not re-emitted.

Consumers that re-emit ingested content for crawling by search engines should take reasonable measures to limit any use of their site as a Search Engine Optimization loophole.

This may include converting un-trusted hyperlinks to text or including a `rel="nofollow"` attribute.

Consumers should be aware of the potential for spoofing attacks where the attacker publishes activities or objects with falsified property values with the intent of injecting malicious content, hiding or corrupting legitimate content, or misleading users.

Activity Streams are JSON Documents and are subject to the same security considerations described in [\[RFC4627\]](#).

Activity Streams implementations handle URIs. See Section 7 of [\[RFC3986\]](#).

Activity Streams implementations handle IRIs. See Section 8 of [\[RFC3987\]](#).

12. IANA Considerations

TOC

None.

13. License

TOC

As of [date], the following persons or entities have made this Specification available under the Open Web Foundation Agreement Version 1.0, which is available at <http://www.openwebfoundation.org/legal/>.

[List of persons or entities]

You can review the signed copies of the Open Web Foundation Agreement Version 1.0 for this Specification at <http://activitystrea.ms/licensing/>, which may also include additional parties to those listed above.

Your use of this Specification may be subject to other third party rights. THIS SPECIFICATION IS PROVIDED "AS IS." The contributors expressly disclaim any warranties (express, implied, or otherwise), including implied warranties of merchantability, non-infringement, fitness for a particular purpose, or title, related to the Specification. The entire risk as to implementing or otherwise using the Specification is assumed by the Specification implementer and user. IN NO EVENT WILL ANY PARTY BE LIABLE TO ANY OTHER PARTY FOR LOST PROFITS OR ANY FORM OF INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER FROM ANY CAUSES OF ACTION OF ANY KIND WITH RESPECT TO THIS SPECIFICATION OR ITS GOVERNING AGREEMENT, WHETHER BASED ON BREACH OF CONTRACT, TORT (INCLUDING NEGLIGENCE), OR OTHERWISE, AND WHETHER OR NOT THE OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

14. Errata

TOC

Changes and clarifications to this document are available at <http://activitystrea.ms/specs/json/1.0/errata>.

15. Normative References

TOC

- [RFC2119]** Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels," March 1997.
- [RFC3339]** Klyne, G., "Date and Time on the Internet: Timestamps," July 2002.

- [RFC3986]** Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI)," January 2005.
- [RFC3987]** Duerst, M. and M. Suignard, "Internationalized Resource Identifiers (IRIs)," January 2005.
- [RFC4627]** Crockford, D., "The application/json Media Type for JavaScript Object Notation (JSON)," July 2006.

TOC

Appendix A. Acknowledgements

The authors wish to thank the Activity Streams community and implementers for their support, encouragement, and enthusiasm including but not limited to: Abdul Qabiz, Adina Levin, Adrian Chan, Adriana Javier, Alan Hoffman, Alex Kessinger, Alexander Ovchinnikov, Alexander Zhuravlev, Alexandre Loureiro Solleiro, Amy Walgenbach, Andres Vidal, Angel Robert Marquez, Ari Steinberg, Arjan Scherpenisse, Arne Roomann-Kurrik, Beau Lebens, Ben Hedrington, Ben Metcalfe, Ben Werdmuller, Benjamin Goering, Bill de hÓra, Bo Xing, Bob Aman, Bob Wyman, Brett Slatkin, Brian Walsh, Brynn Evans, Charlie Cauthen, Chris Chabot, Chris Toomey, Christian Crumlish, Dan Brickley, Dan Scott, Daniel Chapman, Danny Ayers, Dare Obasanjo, Darren Bounds, David Cramer, David Nelson, David Recordon, DeWitt Clinton, Douglas Pearce, Ed Summers, Elias Bizannes, Elisabeth Norris, Eric Marcoullier, Eric Woods, Evan Prodromou, Gee-Hsien Chuang, Greg Biggers, Gregory Foster, Henry Saputra, Hillary Madsen, Howard Liptzin, Hung Tran, Ian Kennedy, Ian Mulvany, Ivan Pulleyn, Jacob Kim, James Falkner, James Pike, James Walker, Jason Kahn, Jason Kantz, Jeff Kunins, Jeff Martin, Jian Lin, Johannes Ernst, John Panzer, Jon Lebkowsky, Jon Paul Davies, Jonathan Coffman, Jonathan Dugan, Joseph Boyle, Joseph Holsten, Joseph Smarr, Josh Brewer, Jud Valeski, Julien Chaumond, Julien Genestoux, Jyri Engeström, Kaliya Hamlin, Kevin Marks, Laurent Eschenauer, Laurie Voss, Leah Culver, Libby Miller, Manu Mukerji, Mark Weitzel, Marko Degenkolb, Marshall Kirkpatrick, Martin Svensson, Marty Alchin, Mary Hoder, Matt Leventi, Matt Wilkinson, Matthias Müller-Prove, Max Engel, Max Wegmueller, Melvin Carvalho, Michael Buckbee, Michael Chan, Michael Richardson, Michael Sullivan, Mike Macgirvin, Mislav Marohnić, Mo Jangda, Nate Benes, NeilFred Picciotto, Nick Howard, Nick Lothian, Nissan Dookeran, Nitya Narasimhan, Pablo Martin, Pádraic Brady, Pat G. Cappalaere, Patrick Aljord, Peter Ferne, Peter Reiser, Peter Saint-Andre, Phil Wolff, Philip (flip) Kromer, Richard Cunningham, Richard Zhao, Rick Severson, Robert Hall, Robert Langbert, Robin Cover, Ryan Boyd, Sam Sethi, Scott Raymond, Scott Seely, Simon Grant, Simon Wistow, Stephen Garcia, Stephen Sisk, Stephen Paul Weber, Steve Ivy, Steve Midgley, Steven Livingstone-Perez, Sylvain Carle, Sylvain Hellegouarch, Tantek Çelik, Tatu Saloranta, Tim Moore, Timothy Young, Todd Barnard, Tosh Meston, Tyler Gillies, Zach Copley, and Zach Shepherd.

TOC

Authors' Addresses

James M Snell
IBM

Martin Atkins
SAY Media

Will Norris
Google

Chris Messina
Citizen Agency, Google

Monica Wilkinson
MySpace, Facebook, VMware

Rob Dolin
Microsoft