The iCalendar VINSTANCE Component

Working Draft Standard

Warning for drafts
This document is not a CalConnect Standard. It is distributed for review and comment, and is subject to change without notice and may not be referred to as a Standard. Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation. Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

The Calendaring and Scheduling Consortium, Inc. 2016

Contents

Abstract

This document updates the iCalendar ([IETF RFC 5545](#)) specification to allow a more compact representation of overridden recurrence instances.

Introduction

The iCalendar [IETF RFC 5545](#) data format is in widespread use to represent calendar data. iCalendar has a data model that supports the concept of recurring, or repeating, events (or other types of objects such as tasks). With repeating events, it is often the case that one particular instance may differ from the rest. In that case iCalendar requires that all the data for that event be included, even though only a small portion of it may be different. For long lived recurring events with lots of attendees present, this can often result in a significant increase in the size of the iCalendar data as many instances get overridden.

This specification updates the iCalendar data model to support a new iCalendar component that can be used to represent just the changes in an overridden instance, rather than having to include everything describing it. This can significantly reduce the size of the iCalendar data, leading to reductions in network I/O (with resultant savings in battery usage on mobile devices), and storage requirements, on clients, servers, and associated databases.

The iCalendar VINSTANCE Component

1. Scope

This document updates the iCalendar (IETF RFC 5545) specification to allow a more compact representation of overridden recurrence instances.

2. Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IETF RFC 2119, S. BRADNER. *Key words for use in RFCs to Indicate Requirement Levels*. 1997. RFC Publisher. https://www.rfc-editor.org/info/rfc2119.

IETF RFC 4791, C. DABOO, B. DESRUISSEAUX and L. DUSSEAULT. *Calendaring Extensions to WebDAV (CalDAV)*. 2007. RFC Publisher. https://www.rfc-editor.org/info/rfc4791.

IETF RFC 5234, P. OVERELL. *Augmented BNF for Syntax Specifications: ABNF*. 2008. RFC Publisher. https://www.rfc-editor.org/info/rfc5234.

IETF RFC 5545, B. DESRUISSEAUX (ed.). *Internet Calendaring and Scheduling Core Object Specification (iCalendar)*. 2009. RFC Publisher. https://www.rfc-editor.org/info/rfc5545.

IETF RFC 5546, C. DABOO (ed.). *iCalendar Transport-Independent Interoperability Protocol (iTIP)*. 2009. RFC Publisher. https://www.rfc-editor.org/info/rfc5546.

3. Terms and definitions

No terms and definitions are listed in this document.

4. Terms and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in IETF RFC 2119.

The notation used in this memo is the ABNF notation of IETF RFC 5234 as used by iCalendar IETF RFC 5545. Any syntax elements shown below that are not explicitly defined in this specification come from iCalendar IETF RFC 5545, CalDAV IETF RFC 4791, and [draft-patch].

5. Overview

Recurring events (or other types of component) in iCalendar are defined by the presence of "RRULE", "RDATE", and "EXDATE" properties in a "master component". Those rules produce a set of "generated instances". "Generated instances" do not need to have a representation in the iCalendar data, as their content can be inferred from the "master component". In some cases specific "generated instances" are changed, resulting in an "overridden instance". An "overridden instance" is represented in iCalendar as an "overridden component", which has the same "UID" property value as the "master component", and a "RECURRENCE-ID" property whose value matches the start time of the corresponding "generated instance" (which can be different from the actual start time of the "overridden instance").

For example, consider the following daily recurring event, with no overridden instances. This event defines a set of generated instances for 20160902, 20160903, 20160904, etc.

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VEVENT
UID:1234
DTSTART;VALUE=DATE:20160902
DURATION:PT1H
SUMMARY:Master component
LOCATION:My office
RRULE:FREQ=DAILY
END:VEVENT
END:VCALENDAR
```

If the summary of the second instance needs to be changed, the resulting iCalendar object would be:

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VEVENT
UID:1234
DTSTART;VALUE=DATE:20160902
DURATION:PT1H
SUMMARY:Master component
LOCATION:My office
RRULE:FREQ=DAILY
END:VEVENT
BEGIN:VEVENT
UID:1234
RECURRENCE-ID;VALUE=DATE:20160903
DTSTART;VALUE=DATE:20160903
DURATION:PT1H
SUMMARY:Override second instance
LOCATION:My office
END:VEVENT
END:VCALENDAR
```

As can be seen, the overridden component for 20160903 duplicates many iCalendar properties from the master component, with the "SUMMARY" property being different, "RECURRENCE-ID" added, and "DTSTART" adjusted to the start time of the instance. All the other properties are the same as the corresponding ones in the master component. Using the new representation described by this specification, this iCalendar object would appear as:

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VEVENT
UID:1234
DTSTART;VALUE=DATE:20160902
DURATION:PT1H
SUMMARY:Master component
LOCATION:My office
RRULE:FREQ=DAILY
BEGIN:VINSTANCE
RECURRENCE-ID;VALUE=DATE:20160903
SUMMARY:Override second instance
END:VINSTANCE
END:VEVENT
```

`END:VCALENDAR`

In this case the "VINSTANCE" component is used to encapsulate just those iCalendar properties that are different from the generated instance — just "SUMMARY" in this case. The new representation is just over 80% of the size of the traditional representation, giving a small, but not insignificant, reduction in size. But as more overrides are needed, or as more properties are added to the master component, the savings will increase dramatically.

6. VINSTANCE Component

This specification defines the new "VINSTANCE" component, which is used to represent just the differences between a generated instance, derived from the master component, and the actual overridden instance that will be used instead of the generated instance.

"VINSTANCE" components MUST only appear as sub-components within master components (e.g., they can only appear as sub-components of components that include at least one of either "RRULE" or "RDATE" properties).

Each "VINSTANCE" component MUST include a "RECURRENCE-ID" property with a value that identifies the instance being overridden. Multiple "VINSTANCE" components can appear within a single master component, but each MUST have a unique "RECURRENCE-ID" property value. Unlike traditional overridden components, "VINSTANCE" components MUST NOT include a "UID" property (the effective "UID" is that of the master component enclosing the "VINSTANCE" component).

The iCalendar components and properties within a "VINSTANCE" component are interpreted as follows:

1) Any component appearing in "VINSTANCE" is treated as an addition or update to the corresponding generated instance (see Clause 7).
2) Any property (excluding "RECURRENCE-ID" and "INSTANCE-DELETE") appearing in "VINSTANCE" is treated as an addition or update to the corresponding generated instance (see Clause 8).
3) Any "INSTANCE-DELETE" property appearing in "VINSTANCE" indicates the removal of a component or property from the corresponding generated instance (see Clause 9)

7. Adding or Updating Components in an Overridden Instance

All sub-components in the master component MUST have a "UID" property with a value that is unique (at least within that master component). This allows such sub-components to be easily identified for the purposes of indicating whether a "VINSTANCE" is adding, updating, or deleting a sub-component.

Any iCalendar component defined in the "VINSTANCE" component (referred to here as the "instance component"), other than a "PATCH" component [draft-patch], is treated as either an addition to the generated instance, or as an update of an existing component in the generated instance, as follows:

1) If the generated instance does not contain a sub-component with a "UID" property value matching that of the instance component, then the instance component is an addition to the generated instance.
2) If a sub-component of the generated instance contains a "UID" property with a value that matches that of the instance component, then the instance component replaces the matching one in the generated instance.

Alternatively, a sub-component can be "incrementally" updated by use of a "PATCH" component [draft-patch] that targets the sub-component. A "PATCH" component is created by determining the difference between the sub-component in the overridden instance and the matching sub-component in the generated instance. The "PATCH-TARGET" property value in the "PATCH" component is specified as a path relative to the overridden component targeting the sub-component by its "UID" property value (e.g., if a "VALARM" component inside a

"VEVENT" component is being updated, the "PATCH-TARGET" property value would be "/ VALARM[UID=1234]" (assuming that the "UID" property value of the sub-component being updated is "1234"). The "PATCH" component is then added as a sub-component of the "VINSTANCE" component representing the overridden instance.

When expanding a "VINSTANCE" component into an overridden component, any "PATCH" sub-components are used to update sub-components in the generated instance, by applying the patch processing rules to those sub-components.

8. Adding or Updating Properties in an Overridden Instance

Any iCalendar property (other than "RECURRENCE-ID" and "INSTANCE-DELETE") defined in the "VINSTANCE" component (referred to here as the "instance property") is treated as either an addition to the generated instance, or as an update of an existing property in the generated instance. A "INSTANCE-ACTION" Clause 11.3 property parameter can be defined on instance properties and is used to control how they are processed. The following rules are used to process such properties:

1) If the instance property does not contain an "INSTANCE-ACTION" property parameter, or contains an "INSTANCE-ACTION" property parameter with the default value "BYNAME", then all properties with the same name in the generated instance are replaced by the instance property.
2) If the instance property contains an "INSTANCE-ACTION" property parameter with the value "CREATE", then the instance property is added to the generated instance.
3) If the instance property contains an "INSTANCE-ACTION" property parameter with the value "UPDATE", then all properties with the same name and same value in the generated instance have their parameters updated or removed as follows: .. Any parameter name appended to the "UPDATE" parameter value with a "~" separator is used to remove matching parameters from the derived instance. .. Any parameters, other than "INSTANCE-ACTION", replace any parameters with the same name in the derived instance, or are added to the derived instance if no there is no parameter with the same name.
4) If the instance property contains an "INSTANCE-ACTION" property parameter with the value starting with "BYPARAM", then all properties with the same name and a property parameter that matches the one that is part of the "INSTANCE-ACTION" property value, in the generated instance are replaced by the instance property.

The "INSTANCE-ACTION=BYNAME" operation is used for adding or updating "singleton" properties — properties that only appear once in a given iCalendar component (e.g., "DTSTART", "DTEND", "LOCATION", etc).

The "INSTANCE-ACTION=CREATE" operation is used for adding "multi-occurring" properties — properties that can appear more than once in a given iCalendar component (e.g., "ATTENDEE", "ATTACH", "EXDATE", etc).

The "INSTANCE-ACTION=UPDATE" operation is used for updating parameters on a specific "multi-occurring" property that can be uniquely identified by its value (e.g., the "ATTENDEE" property can appear multiple times in a "VEVENT" component, but each property will have a unique value in that component). This operation cannot be used when the value of the property is being changed. Instead, the "INSTANCE-ACTION=BYPARAM" operation can be used to identify the property being replaced.

The "INSTANCE-ACTION=BYPARAM" operation is used for updating a specific "multi-occurring" property that can be uniquely identified by a parameter value that is the same as one in the instance property.

There may be some situations where a multi-occurring property cannot be uniquely identified. In such cases, the solution to updating one or more of them is to use an "INSTANCE-ACTION=BYNAME" to replace all the existing properties with one new one, then use "INSTANCE-ACTION=CREATE" to add back others that are unchanged or also being updated. Whilst this is

not ideal, it is anticipated that these situations can be avoided by adding appropriate property parameters with unique values to help disambiguate the multi-occurring properties.

9. Deleting Components or Properties from an Overridden Instance

The "INSTANCE-DELETE" property (defined in Clause 11.2) is used to indicate deletion of iCalendar elements from the generated instance. As such, the value of the "INSTANCE-DELETE" property is always a relative path (see Clause 10) that refers to an element that is an immediate "child" of the generated instance.

The following operations are supported:

Delete components	the "INSTANCE-DELETE" path value identifies components only. The matching components are removed from the generated instance.
Delete properties	the "INSTANCE-DELETE" path value identifies properties only. The matching properties are removed from the generated instance.

10. iCalendar Path

This specification makes use of the concept of an "iCalendar path" defined in Section XX of [draft-patch]. This specification only makes use of "relative" paths to identify components or properties directly within a generated instance.

11. iCalendar Extensions

11.1. VINSTANCE Component

Component Name	VINSTANCE
Purpose	Provide components and properties that are used to indicate the difference between a generated instance and an actual overridden instance.
Format Definition	A "VINSTANCE" calendar component is defined by the following notation:

```
vinstancec    = "BEGIN" ":" "VINSTANCE" CRLF
                  vinstanceprop component
              "END" ":" "VINSTANCE" CRLF
              ; Any sub-component allowed except for
              ; VINSTANCE itself. PATCH sub-components
              ; are used for incremental updates.

vinstanceprop = *(
                  ;
                  ; The following is REQUIRED,
                  ; but MUST NOT occur more than once.
                  ;
                  recurid /
                  ;
                  ; The following are OPTIONAL,
                  ; and MAY occur more than once.
                  ;
                  instancedelete / other-prop
                  ;
                )
```

```
other-prop  = ( iana-prop / x-prop )
```

Description   This component is used to define an overridden instance of a master component that can include changes to the generated instance, such as component or property additions updates, or deletions. See [Clause 6](#) for details.

11.2. INSTANCE-DELETE Property

Property Name   INSTANCE-DELETE

Purpose   This property specifies a relative path identifying one or more components or properties to be removed from a generated instance.

Value Type   TEXT

Property Parameters   IANA and nonstandard property parameters can be specified on this property.

Conformance   This property can be specified within a "VINSTANCE" component only.

Description   This property is used to match iCalendar components or properties that will be deleted. The path value is always a relative path for only immediate components and properties within the generated instance, and interpreted as described in [Clause 9](#).

Format Definition   This property is defined by the following notation:

```
idelete      = "INSTANCE-DELETE ideleteparam ":"
ideletepath CRLF

ideleteparam  = *(";" other-param)

ideletepath   = comp-path / prop-path
              ; from Section XX of [draft-patch]
```

Example   The following are examples of this property:

```
INSTANCE-DELETE:/VALARM[UID=1234]
INSTANCE-DELETE:#ATTENDEE[=mailto:cyrus@example.com]
```

11.3. INSTANCE-ACTION Property Parameter

Parameter Name   INSTANCE-ACTION

Purpose   To specify whether the property should be added or replaced.

Description   This property parameter can be specified on properties contained in a "VINSTANCE" component and MUST NOT be specified on properties outside of a "VINSTANCE" component. This property parameter specifies whether the associated property should be added to the generated instance or should replace existing properties in the generated instance. In the latter case, the property parameter also specifies how to match existing properties. The processing of this property parameter is described in [Clause 8](#).

Format Definition	This parameter is defined by the following notation:

```
iactionparam   = "INSTANCE-ACTION" "="
                   iactioncreate /
                   iactionupdate /
                   iactionbyname /
                   iactionbyparam /
                   iana-token /     ; IANA registered value
                   x-name           ; Experimental value

iactioncreate  = "CREATE"
                 ; Always add property to the generated
instance.

iactionupdate  = "UPDATE" *["~" param-name]
                 ; Always update or remove parameters on
properties
                 ; with the same name and value in the
generated
                 ; instance.

iactionbyname  = "BYNAME"
                 ; Always replace properties with the same name
                 ; in the generated instance.
                 ; This value is the default and MAY be
omitted.

iactionbyparam = DQUOTE "BYPARAM" param-match  DQUOTE
                 ; Always replace properties with the same name
                 ; and parameter name/value in the generated
                 ; instance.
```

Examples	The following are examples of this property parameter:

```
ATTENDEE;INSTANCE-ACTION=UPDATE~RSVP;PARTSTAT=NEEDS-ACTION:
 mailto:cyrus@example.com
DESCRIPTION;INSTANCE-ACTION="BYPARAM@LANGUAGE=en_GB";LANGUAGE=
en_US:
 Meeting to discuss VINSTANCE
```

12. Conversion to/from VINSTANCE

Any iCalendar processing engine that supports "VINSTANCE" is allowed to convert between the traditional overridden component representation and the "VINSTANCE" based representation when processing iCalendar data. However, iCalendar data SHOULD NOT mix traditional and "VINSTANCE" based representations in the same iCalendar object. However, when iCalendar data is transferred to another system whose support for "VINSTANCE" cannot be confirmed, the iCalendar data being transferred MUST be converted into its traditional overridden component representation.

13. Use with iTIP

iTIP IETF RFC 5546 defines how iCalendar data can be sent between calendar user agents to schedule calendar components between calendar users. This specification does not define how iCalendar objects using "VINSTANCE" components can be used with iTIP.

14. Use with CalDAV and HTTP

The CalDAV IETF RFC 4791 calendar access protocol allows clients and servers to exchange iCalendar data.

TBD define protocol changes required in clients and servers to allow negotiated use of VINSTANCE.

15. Use with iCalendar VPATCH

[draft-patch] defines how iCalendar data can updated using a "patch" component that defines the changes between the original and updated data. This generic mechanism works with the new "VINSTANCE" representation introduced by this specification.

However, [draft-patch] does define an "implicit" recurrence override mechanism, whereby a patch operation can implicitly create an overridden component in an iCalendar object by only including components and/or properties that are different from the generated instance. An example of an implicit override patch is shown in Appendix B.1. Whilst it is possible to explicitly add a "VINSTANCE" component using a patch operation (as shown in Appendix B.2), it would be more efficient to allow patch processing engines to create "VINSTANCE" components, rather than full overridden components, when doing an implicit patch operation. This specification extends [draft-patch] to allow patch processing engines to implement an implicit recurrence override patch operation as the addition of appropriate "VINSTANCE" components to the master component (an example is shown in Appendix B.3).

16. Security Considerations

Security considerations described in IETF RFC 5545, and IETF RFC 4791 MUST be adhered to. Since this specification merely defines an alternative representation in iCalendar data, it does not introduce any new security considerations.

17. Privacy Considerations

Privacy considerations described in IETF RFC 5545, and IETF RFC 4791 MUST be adhered to. Since this specification merely defines an alternative representation in iCalendar data, it does not introduce any new privacy considerations.

18. IANA Considerations

18.1. Component Registrations

This document defines the following new iCalendar components to be added to the registry defined in IETF RFC 5545, Section 8.3.1:

Table 1

Component	Status	Reference
VINSTANCE	Current	RFCXXXX, Clause 11.1

18.2. Property Registrations

This document defines the following new iCalendar properties to be added to the registry defined in IETF RFC 5545, Section 8.3.2:

Table 2

Property	Status	Reference
INSTANCE-DELETE	Current	RFCXXXX, Clause 11.2

18.3. Parameter Registrations

This document defines the following new iCalendar parameters to be added to the registry defined in IETF RFC 5545, Section 8.3.3:

Table 3

Property	Status	Reference
INSTANCE-ACTION	Current	RFCXXXX, Clause 11.3

18.4. Parameter Value Registry

A new IANA registry for iCalendar elements has been added. Additional codes MAY be used, provided the process described in IETF RFC 5545, Section 8.2.1 is used to register them, using the template in IETF RFC 5545, Section 8.2.6.

18.4.1. Instance Action Registry

The following table has been used to initialize the Instance Action Registry:

Table 4

Instance Action	Status	Reference
CREATE	Current	RFCXXXX, Clause 11.3
UPDATE	Current	RFCXXXX, Clause 11.3
BYNAME	Current	RFCXXXX, Clause 11.3
BYPARAM	Current	RFCXXXX, Clause 11.3

19. Acknowledgments

Thanks to the following for feedback: Michael Douglass, Ken Murchison.

This specification originated from work at the Calendaring and Scheduling Consortium, which has helped with the development and testing of implementations.

Appendix A
(normative)
Examples

A.1. Overridden instance with just the SUMMARY changed

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VEVENT
UID:1234
DTSTART;VALUE=DATE:20160902
DURATION:PT1H
SUMMARY:Master component
LOCATION:My office
RRULE:FREQ=DAILY
BEGIN:VINSTANCE
RECURRENCE-ID;VALUE=DATE:20160903
SUMMARY:Override second instance
END:VINSTANCE
END:VEVENT
END:VCALENDAR
```

A.2. Overridden instance with a time change and an alarm added

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VEVENT
UID:1234
DTSTART:20160902T120000Z
DURATION:PT1H
SUMMARY:Master component
LOCATION:My office
RRULE:FREQ=DAILY
BEGIN:VINSTANCE
RECURRENCE-ID:20160903T120000Z
DTSTART:20160903T130000Z
BEGIN:VALARM
UID:4567
ACTION:DISPLAY
TRIGGER:-PT30M
DESCRIPTION:Time to leave
END:VALARM
END:VINSTANCE
END:VEVENT
END:VCALENDAR
```

A.3. Overridden instance with a different trigger time for the alarm

This variant shows the "VALARM" being updated by using a "PATCH" component as opposed to include the entire "VALARM" component.

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VEVENT
UID:1234
DTSTART:20160902T120000Z
DURATION:PT1H
```

```
SUMMARY:Master component
LOCATION:My office
RRULE:FREQ=DAILY
BEGIN:VALARM
UID:4567
ACTION:DISPLAY
TRIGGER:-PT30M
DESCRIPTION:Time to leave
END:VALARM
BEGIN:VINSTANCE
RECURRENCE-ID:20160903T120000Z
BEGIN:PATCH
PATCH-TARGET:/VALARM[UID=4567]
TRIGGER:-PT5M
END:PATCH
END:VINSTANCE
END:VEVENT
END:VCALENDAR
```

A.4. Overridden instance without the alarm

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VEVENT
UID:1234
DTSTART:20160902T120000Z
DURATION:PT1H
SUMMARY:Master component
LOCATION:My office
RRULE:FREQ=DAILY
BEGIN:VALARM
UID:4567
ACTION:DISPLAY
TRIGGER:-PT30M
DESCRIPTION:Time to leave
END:VALARM
BEGIN:VINSTANCE
RECURRENCE-ID:20160903T120000Z
INSTANCE-DELETE:/VALARM[UID=4567]
END:VINSTANCE
END:VEVENT
END:VCALENDAR
```

A.5. Two overridden instances with parameter updates

This examples shows two overridden instances. The first one covers the case where an attendee has accepted all but one instance of a recurring meeting, with the one instance being declined. The second one covers the case where an attendee has responded only to one instance of a recurring meeting, with the other instances still waiting for a response.

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VEVENT
UID:1234
DTSTART:20160902T120000Z
DURATION:PT1H
SUMMARY:Master component
LOCATION:My office
RRULE:FREQ=DAILY
```

```
ORGANIZER;CN=Cyrus Daboo:mailto:cyrus@example.com
ATTENDEE;CN=Cyrus Daboo;PARTSTAT=ACCEPTED:
 mailto:cyrus@example.com
ATTENDEE;CN=Mike Douglass;PARTSTAT=NEEDS-ACTION;
 RSVP=TRUE:mailto:mike@example.com
ATTENDEE;CN=Ken Murchison;PARTSTAT=ACCEPTED:
 mailto:ken@example.com
BEGIN:VINSTANCE
RECURRENCE-ID:20160903T120000Z
ATTENDEE;INSTANCE-ACTION=UPDATE;PARTSTAT=DECLINED:
 mailto:ken@example.com
END:VINSTANCE
BEGIN:VINSTANCE
RECURRENCE-ID:20160904T120000Z
ATTENDEE;INSTANCE-ACTION=UPDATE~RSVP;PARTSTAT=ACCEPTED:
 mailto:mike@example.com
END:VINSTANCE
END:VEVENT
END:VCALENDAR
```

Appendix B
(normative)
Patch Examples

B.1. Implicit override using patch

This example shows the implicit addition of an overridden component which has only its
"SUMMARY" property changed, via a patch operation.

Before:

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VEVENT
UID:1234
DTSTART;VALUE=DATE:20160902
DURATION:PT1H
SUMMARY:Master component
LOCATION:My office
RRULE:FREQ=DAILY
END:VEVENT
END:VCALENDAR
```

Patch:

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VPATCH
UID:DFB887F7-9D1A-4FB2-912B-C91A6203DB8E
DTSTAMP:20160907T111100Z
BEGIN:PATCH
PATCH-TARGET:/VCALENDAR/VEVENT[RID=20160903]
SUMMARY:Override second instance
END:PATCH
END:VPATCH
END:VCALENDAR
```

After:

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VEVENT
UID:1234
DTSTART;VALUE=DATE:20160902
DURATION:PT1H
SUMMARY:Master component
LOCATION:My office
RRULE:FREQ=DAILY
END:VEVENT
BEGIN:VEVENT
UID:1234
RECURRENCE-ID;VALUE=DATE:20160903
DTSTART;VALUE=DATE:20160903
DURATION:PT1H
SUMMARY:Override second instance
LOCATION:My office
END:VEVENT
```

```
END:VCALENDAR
```

B.2. Explicit VINSTANCE override using patch

This example shows the explicit addition of an overridden instance using a "VINSTANCE" component, via a patch operation.

Before:

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VEVENT
UID:1234
DTSTART;VALUE=DATE:20160902
DURATION:PT1H
SUMMARY:Master component
LOCATION:My office
RRULE:FREQ=DAILY
END:VEVENT
END:VCALENDAR
```

Patch:

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VPATCH
UID:DFB887F7-9D1A-4FB2-912B-C91A6203DB8E
DTSTAMP:20160907T111100Z
BEGIN:PATCH
PATCH-TARGET:/VCALENDAR/VEVENT
BEGIN:VINSTANCE
RECURRENCE-ID;VALUE=DATE:20160903
SUMMARY:Override second instance
END:VINSTANCE
END:PATCH
END:VPATCH
END:VCALENDAR
```

After:

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VEVENT
UID:1234
DTSTART;VALUE=DATE:20160902
DURATION:PT1H
SUMMARY:Master component
LOCATION:My office
RRULE:FREQ=DAILY
BEGIN:VINSTANCE
RECURRENCE-ID;VALUE=DATE:20160903
SUMMARY:Override second instance
END:VINSTANCE
END:VEVENT
```

```
END:VCALENDAR
```

B.3. Implicit VINSTANCE override using patch

This example shows the implicit addition of an overridden instance using a "VINSTANCE" component, via a patch operation.

Before:

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VEVENT
UID:1234
DTSTART;VALUE=DATE:20160902
DURATION:PT1H
SUMMARY:Master component
LOCATION:My office
RRULE:FREQ=DAILY
END:VEVENT
END:VCALENDAR
```

Patch:

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VPATCH
UID:DFB887F7-9D1A-4FB2-912B-C91A6203DB8E
DTSTAMP:20160907T111100Z
BEGIN:PATCH
PATCH-TARGET:/VCALENDAR/VEVENT[RID=20160903]
SUMMARY:Override second instance
END:PATCH
END:VPATCH
END:VCALENDAR
```

After:

```
BEGIN:VCALENDAR
PRODID:test
VERSION:2.0
BEGIN:VEVENT
UID:1234
DTSTART;VALUE=DATE:20160902
DURATION:PT1H
SUMMARY:Master component
LOCATION:My office
RRULE:FREQ=DAILY
BEGIN:VINSTANCE
RECURRENCE-ID;VALUE=DATE:20160903
SUMMARY:Override second instance
END:VINSTANCE
END:VEVENT
END:VCALENDAR
```