14 Conformance

14.1 Introduction

This clause specifies conformance of:

a) functional implementations of the SRM (14.2),
b) exchange formats that use SRM data structures and associated data types (14.3),
c) language bindings of the SRM API (14.4),
d) applications that use the SRM API (14.5), and
e) specifications that reference this International Standard (14.6).

Functional implementation and exchange format conformance are based on profiles. Profiles are defined in Clause 12. Conformance of an application to a profile is defined in 14.5.

14.2 Functional implementation conformance

14.2.1 Functional accuracy

The computational accuracies of SRF operations are required in determining the (degree of) conformance of functional implementations of the SRM. This clause addresses the computational accuracy requirements for SRF operations.

Computational accuracy requirements are specified as the maximum computational error for an implementation of an SRF operation over a subset of the CS domain of an SRF, termed an accuracy domain. Annex I presents methods used to compute various types of errors. The default profile 12.3 specifies the maximum computational error for three categories of computational error: position, direction, and ratio, as well as accuracy domains for various sets of SRFs included in the profile. The computational accuracy requirement does not apply to a sequence or chain of SRF operations, only to each individual SRF operation in the sequence. This clause does not directly address the software environment, performance, or resource requirements of applications or implementations that conform to profiles of this International Standard. Annex B presents advisory guidelines for implementation efficiency and error control. This clause does not define the application requirements or dictate the functional content of applications that use SRM implementations.

An *accuracy domain* is a set of constraints that specify:

a) a subset of the CS domain for an SRF,
b) the parameter values that realize the CS domain for the SRFs that can be instantiated from an SRFT,
c) the parameter values that are common to a collection of SRFTs.

To be conformant, the computational accuracy requirements need only to be satisfied for coordinates in the accuracy domain. The CS domain subset specified by an accuracy domain shall be non-empty and both closed and bounded (see A.3).

When an accuracy domain is defined for an SRF, it can be specified in one or more of the following ways:

a) constraints on coordinate-component values for one or more coordinate-components,
b) constraints that specify a region of the CS range, which indirectly constrains coordinate-component values to those that have CS generating function values in that region,
    1) the CS range region can also be specified in terms of other functions on the CS domain, such as Euclidean distance,

2) for map projections, the CS range region can be specified in terms of one or more constraints on 3D geodetic CS coordinate-component values,

EXAMPLE 1 For an EQUATORIAL_SPHERICAL SRF an accuracy domain is specified with a value constraint on the radius coordinate component using a closed bounded interval: $0{,}01\text{m} \leq \rho \leq 1\,000\,000\,000\text{m}$.

EXAMPLE 2 For a LOCOCENTRIC _EUCLIDEAN_3D SRF an accuracy domain is specified by restricting the CS range to the region satisfying the restriction: $d_E(\boldsymbol{c}, (0,0,0)) \leq 1\,000\,000\,000\text{m}$ for all $\boldsymbol{c}$ in the CS domain where $d_E(\boldsymbol{c}, (0,0,0))$ is the Euclidean distance function (see 10.6).

EXAMPLE 3 For a TRANSVERSE_MERCATOR SRF with longitude of origin $\lambda_{\text{origin}} = 14{,}4(\pi/180)$, an accuracy domain is specified by restricting the CS range to the region bounded by geodetic constraints:

$$-3{,}5(\pi/180) \leq \lambda - 14{,}4(\pi/180) \leq 3{,}5(\pi/180),$$
$$-89{,}5(\pi/180) \leq \varphi \leq 89{,}5(\pi/180), \text{ and}$$
$$-50\,000\text{m} \leq h \leq +1\,000\,000\text{m}.$$

When an accuracy domain is defined for an SRFT, it applies to any SRF that can be instantiated from that SRFT. In that case, an accuracy domain can specify constraints that depend on SRFT parameter values (if any), as well as on CS coordinate-component values. Individual constraints can include any combination of CS coordinate-component values and/or SRFT parameter values. Constraints including only SRFT parameter values are commonly used to constrain the ORM RDs that can be used by an SRFT.

When an accuracy domain is defined for a collection of SRFTs, the SRFTs in the collection shall share common CS coordinate-components and/or SRFT parameters:

a) each coordinate-component used in a constraint shall be common to all the SRFTs in the collection,

b) each SRFT parameter used in a constraint shall be common to all the SRFTs in the collection.

EXAMPLE 4 For SRFT TRANSVERSE_MERCATOR, an accuracy domain is specified with:

$-3{,}5(\pi/180) \leq \lambda - \lambda_{\text{origin}} \leq 3{,}5(\pi/180),$	(using the SRFT parameter λ_{origin})
$a \leq 6\,400\,000m$ and $f \leq 1/150$, and	(using SRFT parameters a and f)
$0{,}01m \leq k_{\text{o}}$	(using SRFT parameter k_{o})

The *error criteria for operations on SRFs* derived from a given SRFT are determined by an accuracy domain specification together with a set of error bounds. Operations on the SRFs derived from the SRFT satisfy the error criteria if the error at any coordinate in the accuracy domain, determined by the accuracy domain, is less than the error bounds for those operations.

A *computational accuracy requirement* of a profile consists of the error criteria specification for each of the SRFTs belonging to the profile. An implementation conforms to the computational accuracy requirement of a profile if, for each SRFT in the profile, each implemented operation on the SRFs derived from the SRFT satisfies the error criteria for that SRFT.

14.2.2 Functional conformance

A functional implementation of the SRM *conforms* to a standardized or registered profile P, if the following conditions are satisfied:

a) Each SRM concept instance in P shall be identified by the label and code specified for that concept instance in this International Standard or by registration; this includes, but is not limited to, ORMs, SRFTs, SRFs, SRF sets, and DSSs,

b) The implementation shall support the data types required for the API functionality of each of the SRM concept instances in P. Additional functionality and data types may be supported by an implementation. If the implementation supports the API functionality specified in this International Standard, the methods and functions shall use the data types specified in this International Standard.

c) The implementation shall support the full functionality of all operations applicable to each SRM concept instance in P in accordance with the relevant clauses in this international standard.

d) The data types and data structures shall match the specification of the corresponding data types as defined in this International Standard,

e) The units of measure that are used in data structures shall be as specified in this International Standard (see 4.1), and

f) The implementation shall conform to the computational accuracy requirement of profile P (see 12.2).

A functional implementation of the SRM is free to exceed the required conditions of any profile to which it claims conformance. A functional implementation may support additional standardized and/or registered SRM concept instances that are not included in any profile to which it claims conformance, including ORMs, SRFTs, SRFs, SRF sets, and/or DSSs. For any supported SRFTs, a functional implementation may satisfy smaller error bounds than those specified in the computational accuracy requirements for those SRFTs.

A functional implementation that conforms to profile P satisfies an application if all of the concept instances and associated operations that the application references are included in profile P.

14.3 Conformance of exchange formats

An exchange format *conforms* to a standardized or registered profile P, if the following conditions are satisfied:

a) Each SRM concept instance in P shall be identified by the label and/or code specified for that concept instance in this International Standard or by registration; this includes, but is not limited to, ORMs, RTs, SRFTs, SRFs, SRF sets, and DSSs,

b) The data types and data structures shall match the specification of the corresponding data types as defined in this International Standard,

c) All data types and data structures shall be used to represent coordinates in their corresponding SRF as defined in 11.5, and

d) The units of measure that are used in data structures shall be as specified in this International Standard (see 4.1).

14.4 Conformance of language bindings of the SRM API

A language binding of the SRM API to a programming language *conforms* to the SRM, if the following conditions are satisfied:

a) All functions specified in Clause 11, including output values and error conditions, shall be so bound as to present the specified interfaces as closely as possible given the strictures of that programming language,

b) All data types specified in this International Standard shall be represented in that programming language,

c) The resulting language binding shall follow the cultural conventions of that programming language, and

d) The language binding shall provide a mapping of SRM concept instance labels to identifiers and/or constants within the language in such a manner as to maintain the symbolic names of this International Standard as closely as possible within the strictures of the programming language for which the binding is created.

Language bindings are allowed to append additional identification to the beginning or end of SRM concept instance labels as necessary to make the symbolic names corresponding to those labels unique and identifiable as part of the subject language binding.

14.5 Conformance of applications that use the SRM API

An application that uses the SRM API *conforms*, if the following conditions are satisfied:

a) The use of any functionality of the SRM API by the application shall conform to the provisions of Clause 11 as it applies to that functionality,

b) Invocations of the SRM API shall pass all parameters in the required units as specified in 4.1, and

c) All error messages received from the API shall be processed as required by this International Standard (see 11.3.1).

An application conforms to a profile P if all of the concept instances and associated operations included in profile P are fully supported by the application.

If an application conforms to a profile P, it can use any functional implementation of the SRM that also conforms to profile P.

14.6 Conformance of specifications that reference this International Standard

Specifications that reference this International Standard shall also use the data types specified in this International Standard (see 11.5).