Definition 1. Generalized form of ConSpec: Given a global session trace $\mathcal{S}_t$, we say that $\mathcal{S}_t$ satisfies a consistency model $\mathcal{C}$ if there exists a partial order $(\mathcal{O}_{St}, \preccurlyeq)$ over the set $\mathcal{O}_{St}$ comprising operations present in all session traces in $\mathcal{S}_t$, i.e., $\mathcal{O}_{St} = \bigcup_{st \in St} \{o \mid o \in st\}$, such that 1) for every operation o in $\mathcal{O}_{St}$, its output is equal to the one obtained by executing the sequential specification of a linear extension of the operations preceding o in $\preccurlyeq$, and 2) $(\mathcal{O}_{St}, \preccurlyeq)$ obeys E^s, which is an LTL expression restricting $(\mathcal{O}_{St}, \preccurlyeq)$.

1 Deriving ConSpec Specifications From Original Definitions

In this section, we illustrate that the ConSpec specifications of consistency models and original definitions [1, 2, 3] are equivalent. RYW, MR, WFR, MW, Causal and Sequential consistency have been defined by Chockler et al. [1], whereas PC have been defined in [2, 3]. We begin by describing the notations used in the original consistency definitions given by Chockler et al., and how they can be translated to the syntax of ConSpec. According to Chockler et al., a system comprises a group of processes that communicate with each other by invoking read or write operations on a group of objects. Chockler et al. denote a pair of read or write operations on an object x, invoked from the i^{th} process p_i in the system, as $o^1(x)$ and $o^2(x)$, respectively. We use the notation Cl_i for the i^{th} client in the system. The process p_i is an execution thread instantiated by the client Cl_i. The notation σ_i is used by Chockler et al. to denote a local execution composed of a sequence of read and write operations performed by a client Cl_i. Each operation is comprised of an invocation event and a response event, such that the response for each operation follows its corresponding invocation. The notation σ denotes a global execution comprising all such local executions performed by all clients in the system. Chockler et al. use the symbol $\rightarrow$ to denote a *precedence* relationship [4] between two operations, such that an expression $o^1(x) \rightarrow o^2(x)$ implies that an invocation of an operation $o^1(x)$ precedes an invocation of $o^2(x)$ in a given execution. $\xrightarrow{\sigma_i}$ is a specialised form of the precedence operator, where the superscript σ_i is used to restrict the precedence relationship $\rightarrow$ to operations comprised in the particular execution σ_i. According to the above system model, an invocation event of an operation can not occur unless the response event of a preceding operation comprised in the same session (i.e., invoked by the same client) has occurred in an execution. Hence, the expression $o^1(x) \xrightarrow{\sigma_i} o^2(x)$ implies that both the invocation and response for operation $o^1(x)$ precede (i.e., happen before) the invocation and response of operation $o^2(x)$ in an execution sequence σ_i. Additionally, Chockler et al. extend the notation σ_i to define a special-purpose notation $\sigma|i+w$ denoting a partial execution, where $i+w$ implies the restriction of a global execution σ to an execution comprising all operations performed by a given client Cl_i (or process p_i) plus all writes invoked by other clients. Chockler et al. also use the notation S_p to denote an equivalent *legal serialization* for a partial execution $\sigma|i+w$; a legal serialization is a linear sequence of invocation of operations such that each read operation in the sequence returns the result of the last preceding write. A special-purpose operator $\xrightarrow{S_p}$ is used to denote the precedence relation among operations comprised in the legal serialization S_p.

Using the above notations, Chockler et al. state the RYW consistency model as

$$o^1(x) \xrightarrow{\sigma_i} o^2(x) \Rightarrow o^1(x) \xrightarrow{S_p} o^2(x), \quad (1)$$

where S_p is an equivalent legal serialization for a partial execution $\sigma|i+w$ comprising operations invoked by p_i plus writes invoked by other clients. Let inv_1 and inv_2 denote the invocations, and $resp_1$ and $resp_2$ denote the responses of o^1 a o^2, respectively. Since the clients are well-formed (as we defined in the definition Section in the ConSpec paper), the precedence relation $o^1 \xrightarrow{\sigma_i} o^2$ in the original RYW definition by [1] implies that $inv^1(x) \xrightarrow{\sigma_i} resp^1(x)$ and $resp^1(x) \xrightarrow{\sigma_i} inv^2(x)$. Any precedence relation defined on an execution sequence σ_i is equivalent to the exact same precedence relation defined on the session trace st in ConSpec. Thus, replacing references to σ_i with the notation st, the expression $o^1(x) \xrightarrow{\sigma_i} o^2(x)$ can be rewritten as $\Box\ inv^1 \rightarrow \Diamond inv^2$. Further, since clients in ConSpec are well-formed, the invocation and response for a given operation precedes the invocation and response of the next operation in st. Thus, $\Box\ inv^1 \rightarrow \Diamond inv^2$ implies $\Box\ resp_1 \rightarrow \Diamond resp_2$, which, in turn, implies $\Box\ o^1(x) \rightarrow \Diamond o^2(x)$. Hence, we can rewrite a precedence relation $o^1(x) \xrightarrow{\sigma_i} o^2(x)$ found in Chockler et al.'s definitions as $\Box\ o^1(x) \rightarrow \Diamond o^2(x)$. Let $Op^1(x)$ and $Op^2(x)$ be propositional logic variables that indicate whether invocations and responses of operations $o^1(x)$ and $o^2(x)$ have executed (if $Op^1(x)$ and $Op^2(x)$ is TRUE) or not. Then, the precondition reduces to the ConSpec form $\Box\ Op^1(x) \rightarrow \Diamond Op^2(x)$.

In the postcondition for RYW, Chockler et al. specify a precedence relation $o^1(x) \xrightarrow{S_p} o^2(x)$, which restricts a precedence relation among $o^1(x)$ and $o^2(x)$ in the equivalent legal serialization S_p for the given $\sigma|i+w$. By definition of S_p in the Definition Section in our ConSpec, both the invocation and response for an operation $o^2(x)$ in S_p must appear after the invocation and response of a preceding operation $o^1(x)$ in S_p, i.e., $o^1(x) \xrightarrow{S_p} o^2(x)$ implies $inv^1 \xrightarrow{S_p} inv^2$ and $resp^1 \xrightarrow{S_p} resp^2$. Hence, all components of operation $o^2(x)$ follow all components of $o^1(x)$ in S_p. By the definition of a valid partial order in Definition 1, we can replace S_p in the postcondition with the notation $\preccurlyeq_{st+w}$ (defined in Definition 1 and the RYW Equation in ConSpec paper because of the following reason. Thus, we can rewrite the above precedence relation $o^1(x) \xrightarrow{S_p} o^2(x)$ as $Op^1(x) \preccurlyeq_{st+w} Op^2(x)$. Since RYW considers only those execution sequences where a write operation is followed by a read, $Op^1(x)$ and $Op^2(x)$ can be replaced by new propositional variables $W^{'}(x)$ and $R^{''}(x)$, without any loss of information. Thus, the above precondition and postcondition can be expressed as $\Box\ W^{'}(x) \rightarrow \Diamond R^{''}(x)$ and $\Box\ W^{'}(x) \preccurlyeq_{st+w} R^{''}(x)$, respectively. According to Chockler et al., a legal serialization S_p is a sequence of operations that satisfies the following properties: Property 1) it is a linear sequence that comprises all operations from a given client Cl_i plus writes from all other clients, and Property 2) each read in the sequence S_p returns the result of the preceding write in S_p. First, by definition (refer to definition of $\preccurlyeq$ in the RYW expression, $\preccurlyeq_{st+w}$ is a partial order comprising all operations in a session trace st plus writes from other operations, thus $\preccurlyeq_{st+w}$ satisfies Property 1 for S_p. Second, following directly from precondition of RYW, the output of each operation in $\preccurlyeq_{st+w}$ is equivalent to that obtained by executing a linear sequence of the operations preceding that operation, thus $\preccurlyeq_{st+w}$ satisfies Property 2 for S_p. Hence, we can express the postcondition

$\Box\ W^{'}(x) \preccurlyeq_{st+w} R^{''}(x)$ as $W^{'}(x) \preccurlyeq_{st+w} R^{''}(x)$. Combining the above conditions, Chockler's definition of RYW reduces into the ConSpec specification in the RYW Equation.

According to Chockler et al., MR is expressed in terns of a correctness condition

$$Condition1 \Rightarrow Condition2, \tag{2}$$

where both $o^1(x)$ and $o^2(x)$ are read operations. Following the same logic as that used in the derivation of RYW (refer to the ConSpec specification Section in the ConSpec paper), the precedence relationships among operations $o^1(x)$ and $o^2(x)$ in Condition 1 can be directly expressed in terms of an LTL expression $\Box\ R^{'}(x) \rightarrow \Diamond R^{''}(x)$. Similarly, the expression $o^1(x) \xrightarrow{S_p} o^2(x)$ in Condition 2 can be expressed in terms of a valid partial order $\preccurlyeq_{st+w}$ over st Further, similar to the derivation of RYW, we can rewrite the expression as $R^{'}(x) \preccurlyeq_{st+w} R^{''}(x)$, thus reducing the above specification into the MR Equation in ConSpec paper.

In their definition of Causal consistency, Chockler uses the notion of a *direct precedence relation* between operations $o(x)$ and $o^{'}(x)$ in an execution order σ_i, denoted as $\overset{\sigma_i}{\Rightarrow}$. The expression $o(x) \overset{\sigma_i}{\Rightarrow} o^{'}(x)$ implies that either of the following properties must hold: Property 1) $o^{'}(x)$ is a read operation which returns the values written by a write operation $o(x)$, or Property 2) the precedence relation $o(x) \xrightarrow{\sigma_i} o^{'}(x)$ holds for a given execution σ_i. Causal consistency is expressed as Equation 2. Condition 1 specifies that a transitive closure $\overset{\star}{\Rightarrow}$ exists over a direct precedence relation $o(x) \overset{\sigma_i}{\Rightarrow} o^{'}(x)$ among a given pair of operations $o(x)$ and $o^{'}(x)$ in σ_i. Following the same line of reasoning as that used in our derivation for RYW, Condition 2 can be restated as: there must exist a partial order $\preccurlyeq$ which respects the order specified among the operations performed by each client, i.e., with respect to each observed session trace st, hence, with respect to the global session trace $\mathcal{S}_t$. Thus, Condition 2 reduces to the form $O^{'}(x) \preccurlyeq O^{''}(x)$, where $\preccurlyeq$ is a partial order with respect to operations in the global session trace. As in previous cases, the expression $o(x) \overset{\sigma_i}{\Rightarrow} o^{'}(x)$ in Condition 1 can be expressed in the form $\Box\ O^{'}(x) \rightarrow \Diamond O^{''}(x)$. However, Condition 2 implies that read operation $o(x)v^{'}$, corresponding to the propositional variable $O(x)^{'}$, reads the value written by the write $o(x,v)^{''}$, corresponding to the propositional variable $O^{''}(x)$. Thus, the precondition, comprising a logical disjunction over Condition 1 and 2, can be expressed as $\Box\ O^{'}(x) \rightarrow \Diamond O^{''}(x) \vee \Big(\Big(O^{'}(x) = W^{'}(x)\Big) \wedge \Big(O^{''}(x) = R^{''}(x)\Big) \wedge (v_i = v_j)\Big)$, where $R^{''}(x)$ and $W^{'}(x)$ are shortcut notations for $o(x)v^{'}$ and $o(x,v)^{''}$, respectively. For a given $\mathcal{S}_t$ to satisfy causal consistency, a transitive closure must exist over the above condition. However, it directly follows from the Condition 1 in Definition 1 that if Condition 2 holds, i.e., if a valid $\preccurlyeq$ comprising $o(x)$ and $o^{'}(x)$ exists, every operation in $\preccurlyeq$ must reflect a result which is equivalent to that of executing the prior operations in $\preccurlyeq$ according to a linear sequence. Hence, $\Box\ O^{'}(x) \rightarrow \Diamond O^{''}(x)$ implies that the transitivity condition holds over the expression $\Box\ O^{'}(x) \rightarrow \Diamond O^{''}(x)$ in Condition 1. Hence, the precondition for Causality can simply be expressed as $\Box\ O^{'}(x) \rightarrow \Diamond O^{''}(x)$. Thus, Chockler's definition of Causal Consistency reduces into the specification given in the Causal consistency Equation in ConSpec paper.

Chockler et al. states Sequential Consistency as: the precedence order among operations in a valid legal serialization for a given global execution must match the precedence order of the operations in the local execution of each process, i.e., $o^1 \xrightarrow{\sigma_i} o^2 \Rightarrow o^1 \xrightarrow{S} o^2$, where S in an equivalent legal serialization for the global session execution σ. Following the same approach as in previous derivation, the precedence relation $o^1 \xrightarrow{\sigma_i} o^2$ in the LHS of the above expression can be restated as $\Box\ O^{'} \rightarrow \Diamond O^{''}$. As in our previous derivations, the above RHS can be rewritten as $O^{'}(x) \preccurlyeq O^{''}(x)$, where $\preccurlyeq$ is a partial order comprising all operations in $\mathcal{S}_t$. Thus, $\preccurlyeq$ comprises all operations in the global execution σ. Hence, the condition in the RHS implies a total order $\prec$ among each pair of operations o^1 and o^2 comprised in $\mathcal{S}_t$. Thus, we can replace the partial order symbol $\preccurlyeq$ with $\prec$. This does not cause any loss of information since a total order is a special case of a partial order, i.e., $O^{'}(x) \prec O^{''}(x)$ implies $\Big(O^{'}(x) \preccurlyeq O^{''}(x)\Big) \vee \Big(O^{''}(x) \preccurlyeq O^{'}(x)\Big)$. Hence, we can rewrite the above RHS as $O^{'}(x) \prec O^{''}(x)$. Thus, Chockler's definition of SC reduces into the specification given in the Sequential consistency Equation in ConSpec paper.

References

[1] G. Chockler, R. Friedman, and R. Vitenberg, *Consistency Conditions for a CORBA Caching Service*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 374–388. [Online]. Available: http://dx.doi.org/10.1007/3-540-40026-5_25

[2] M. Ahamad, R. A. Bazzi, R. John, P. Kohli, and G. Neiger, "The power of processor consistency," in *Proceedings of the Fifth Annual ACM Symposium on Parallel Algorithms and Architectures*, ser. SPAA '93. New York, NY, USA: ACM, 1993, pp. 251–260. [Online]. Available: http://doi.acm.org/10.1145/165231.165264

[3] S. Owens, S. Sarkar, and P. Sewell, "A better x86 memory model: X86-tso," in *Proceedings of the 22Nd International Conference on Theorem Proving in Higher Order Logics*, ser. TPHOLs '09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 391–407. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-03359-9_27

[4] P. Bailis, A. Ghodsi, J. M. Hellerstein, and I. Stoica, "Bolt-on causal consistency," in *SIGMOD '13*.